

Copyright
by
Ampol Karoonsoontawong
2006

**The Dissertation Committee for Ampol Karoonsoontawong
certifies that this is the approved version of the following dissertation:**

**Robustness Approach to the Integrated Network Design Problem,
Signal Optimization and Dynamic Traffic Assignment Problem**

Committee:

S. Travis Waller, Supervisor

Randy B. Machemehl

Kara M. Kockelman

David P. Morton

Anant Balakrishnan

**Robustness Approach to the Integrated Network Design Problem,
Signal Optimization and Dynamic Traffic Assignment Problem**

by

Ampol Karoonsoontawong, B.Eng.; M.S.

Dissertation

Presented to the Faculty of the Graduate School of
The University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

The University of Texas at Austin

December 2006

Acknowledgments

The author would like to extend his most profound gratitude to his supervising professor, Dr. S. Travis Waller, for his time, advice, encouragement, and support throughout the course of this study. Thanks and gratitude are also due to Drs. Randy B. Machemehl and Kara M. Kockelman of the Department of Civil Engineering, Dr. David P. Morton of the Operations Research and Industrial Engineering program and Dr. Anant Balakrishnan of the Department of Information, Risk, and Operations Management for their invaluable comments and suggestions.

This dissertation has been completed in conjunction with National Science Foundation (NSF) projects and a Texas Department of Transportation (TxDOT) research project. I would like to gratefully acknowledge the funding provided for this project by NSF and TxDOT. I would like to thank my friends Feng Hong, Feng Wang, Wei Fan, Dung-Ying Lin, Satish Ukkusuri, Deepak Kumar, Avinash Unnikrishnan, Jennifer Duthie, Stephen Boyles, Natalia Ruiz Juri, Lauren Gardner, David Fajardo, and others for discussions and help over the course of my study. Thanks are also due to Chandra Lowmes for her clerical support.

Finally, I would like to thank my parents, Mr. Somsak and Mrs. Anchalee, my sister, Miss Usanee, my brother, Mr. Bancha, and other family members for their moral support.

Robustness Approach to the Integrated Network Design Problem, Signal Optimization and Dynamic Traffic Assignment Problem

Publication No. _____

Ampol Karoonsoontawong, Ph.D.
The University of Texas at Austin, 2006

Supervisor: S. Travis Waller

This dissertation focuses on formulating robust optimization models and developing exact algorithms and various metaheuristics for the integrated network design (NDP), signal setting design (SSD) and dynamic traffic assignment (DTA) problem (NDP-SSD) when accounting for the bi-level nature and the long-term origin-destination demand uncertainty. The NDP determines the optimal budget allocation to improve link capacity, given the available budget. The SSD determines the optimal signal setting (cycle lengths, phase sequencing, green splits and time offsets). NDP-SSD provides a mutually consistent solution where the traffic flows are at dynamic user equilibrium, signal settings are optimal, and link capacity improvement decisions are most favorable. Three bi-level NDP, SSD and NDP-SSD models are proposed: deterministic, stochastic and robust. Also, three single-level SSD and NDP-SSD models are developed: user-optimal, stochastic system-optimal and stochastic user-optimal. The stochastic models minimize the expected cost, whereas the robust models minimize the tradeoff between the expected cost and risk. A solution of robust optimization remains close to optimal for all demand scenarios, while a solution of stochastic optimization may yield large changes in the objective value among different scenarios.

The proposed exact solution methods for bi-level NDP models are the K^{th} -best algorithm and the Karush-Kuhn-Tucker based mixed-integer programming reformulation. For the exact solution methods of bi-level SSD and NDP-SSD models, we show two mathematical proofs for reducing a mixed-zero-one continuous linear bi-level program and a mixed-zero-one continuous quadratic-linear bi-level program to a parametric linear bi-level program and a parametric quadratic-linear bi-level program, respectively. Although the analytical approach is limited to solving a small single-destination network, it is mathematically tractable and provides insight into the problems. For metaheuristic methods, many assumptions of analytical approach can be relaxed, and multi-destination problems can be solved with the use of existing simulation-based DTA software. Four metaheuristics are developed: random search, simulated annealing, genetic algorithm and reactive tabu search.

Extensive numerical experiments are conducted to assess the performance of the metaheuristic algorithms, demonstrate the worthiness of the proposed robust formulations and show the benefit of the integrated approach over the sequential approach. The proposed models and metaheuristic algorithms are tested on limited transportation networks.

Table of Contents

Chapter 1. Introduction.....	1
1.1 Motivation.....	1
1.2 Dissertation Contributions.....	2
1.3 Dissertation Structure.....	6
Chapter 2. Literature Review.....	7
2.1 Network Design Problem.....	7
2.2 Traffic Assignment-Based Signal Optimization	11
2.3 Combined Network Design, Signal Optimization and Traffic Assignment....	13
2.4 Dynamic Traffic Assignment.....	13
2.4.1 Cell Transmission Model.....	14
2.4.2 Dynamic Traffic Assignment Linear Programming Models.....	18
2.4.2.1 System Optimal Dynamic Traffic Assignment.....	18
2.4.2.2 User Optimal Dynamic Traffic Assignment.....	22
2.4.2.3 Concise Formulations of DTA LP Models.....	23
2.4.3 Simulation-based User Optimal Dynamic Traffic Assignment.....	26
2.5 Stochastic Programming.....	29
2.6 Robust Optimization.....	32
2.7 Linear Bi-level Optimization.....	33
2.8 Zero-One Continuous Linear Bi-level Programming.....	35
2.9 Metaheuristics.....	37
2.10 Summary.....	40
Chapter 3. Analytical Approach.....	42
3.1 NDP Models.....	44
3.1.1 Deterministic NDP Models (SONDP, UONDP and BLPNDP).....	45
3.1.2 Two-Stage Stochastic NDP Models (SLP2-SONDP, SLP2-UONDP and SLP2-BLPNDP).....	48
3.1.3 Robust NDP Model (RO-BLPNDP)	52
3.2 Combined DTA and SSD Models.....	54
3.2.1 Deterministic SSD Models (SOSSD, UOSSD and BLPSSD).....	55
3.2.2 Two-Stage Stochastic SSD Models (SLP2-SOSSD, SLP2-UOSSD and SLP2-BLPSSD).....	60
3.2.3 Robust SSD Model (RO-BLPSSD).....	62
3.3 Integrated NDP, SSD and DTA Problem (NDP-SSD).....	63
3.3.1 Deterministic NDP-SSD Models (SONDP-SSD, UONDP-SSD and BLPNDP-SSD).....	64
3.3.2 Two-Stage Stochastic NDP-SSD Models (SLP2-SONDP-SSD, SLP2-UONDP-SSD and SLP2-BLPNDP-SSD).....	66
3.3.3 Robust NDP-SSD Model (RO-BLPNDP-SSD).....	69
3.4 Exact Solution Methods for NDP Models.....	70
3.4.1 Exact Solution Methods for BLPNDP.....	70

3.4.1.1 K^{th} -Best Algorithm.....	71
3.4.1.2 KKT-Based MIP Reformulation.....	75
3.4.2 Exact Solution Method for SLP2-BLPNDP.....	76
3.4.3 Exact Solution Method for RO-BLPNDP.....	79
3.5 Exact Solution Methods for SSD and NDP-SSD Models.....	80
3.5.1 Exact Solution Method for BLPSSD and BLPNDP-SSD.....	80
3.5.2 Exact Solution Method for SLP2-BLPSSD and SLP2-BLPNDP-SSD.....	82
3.5.3 Exact Solution Method for RO-BLPSSD and RO-BLPNDP-SSD.....	83
3.6 Summary.....	84
Chapter 4. Metaheuristic Approach.....	86
4.1 Decision Variables.....	86
4.1.1 Decision Variables of NDP.....	86
4.1.2 Decision Variables of SSD.....	87
4.1.2.1 Three-Leg Intersection.....	88
4.1.2.2 Four-Leg Intersection.....	90
4.1.2.3 Five-leg Intersection.....	94
4.1.3 Decision Variables of NDP-SSD.....	97
4.2 Solution Representations and Decoding Procedures.....	97
4.2.1 Solution Representations and Decoding Procedures for NDP.....	98
4.2.2 Solution Representations and Decoding Procedures for SSD.....	101
4.2.3 Solution Representations and Decoding Procedures for NDP-SSD.....	102
4.3 Objective Functional Evaluation.....	104
4.4 Simulated Annealing (SA).....	105
4.5 Modified Simulated Annealing (MSA).....	110
4.6 Genetic Algorithm (GA).....	111
4.7 Reactive Tabu Search (RTS)	118
4.8 Random Search (RS).....	130
4.9 Summary.....	131
Chapter 5. Experimental Results of Analytical Approach.....	132
5.1 Experimental Results of NDP Models.....	132
5.1.1 Cell Transmission Test Network (Network 1).....	133
5.1.2 Preliminary Results of Deterministic NDP Models.....	134
5.1.2.1 K^{th} Best Algorithm.....	134
5.1.2.2 KKT-Based MIP Reformulation.....	136
5.1.3 Comparison of Seven NDP Models.....	137
5.1.4 Comparison of Robust NDP Based on Moving and Fixed Target TSTT... ..	146
5.2 Experimental Results of the Integrated NDP, SSD and DTA Models.....	151
5.2.1 Cell Transmission Test Network (Network 2).....	151
5.2.2 Comparison of Seven NDP-SSD Models.....	155
5.2.3 Comparison of the Sequential NDP and SSD Approach against the Integrated NDP-SSD Approach.....	161
5.3 Summary.....	164

Chapter 6. Experimental Results of Metaheuristic Approach.....	167
6.1 Test Networks.....	167
6.2 Implementation and Calibration of SA, MSA and GA Parameters.....	172
6.3 Preliminary Performance Comparisons of Metaheuristics for Deterministic NDP on Networks 1, 3.1 and 4.1.....	174
6.3.1 Preliminary Performance Comparison of Metaheuristics for Deterministic NDP on Network 1.....	174
6.3.2 Preliminary Performance Comparisons of SA, GA, and RS for Deterministic NDP on Networks 3.1 and 4.1.....	179
6.4 Comprehensive Performance Comparisons of RS, MSA, GA, RTS-MT0, RTS-MT1 and RTS-MT2 on Networks 3.2 and 4.2.....	182
6.5 Robustness Evaluation of Deterministic Stochastic and Robust Solutions on Networks 3.2 and 4.2.....	193
6.6 Summary.....	208
Chapter 7. Extensions, Summary and Conclusions.....	211
7.1 Extensions.....	211
7.1.1 Dantzig-Wolfe Decomposition-based Heuristic Scheme for BLPNDP..	211
7.1.2 Models Accounting for O-D Demand Elasticity.....	218
7.1.3 Other Metaheuristics.....	218
7.2 Summary and Conclusions.....	219
Appendices.....	226
A. Reduction of MZCL-BLPP to L-BLPP.....	226
B. Reduction of MZC-QL-BLPP to QL-BLPP.....	229
C. Detailed Solutions of NDP Models in Section 5.1.3.....	233
D. Detailed Solutions and Robustness Evaluation Results in Section 5.1.4.....	236
E. Detailed Solutions of Combined NDP, SSD and DTA Models in Section 5.2.2.....	239
F. Factors of O-D Trip Matrix for Networks 4.1 and 4.2; and O-D Demand Realizations for Metaheuristics and Evaluation for Networks 3.2 and 4.2.....	243
G. Detailed Solutions from Metaheuristics in Section 6.5.....	249
H. Robustness Evaluation Results of Deterministic, Stochastic and Robust Solutions from Metaheuristics in Section 6.5.....	254
I. Paired t Tests in Section 6.5.....	259
References.....	271
Vita.....	278

Chapter 1. Introduction

1.1 Motivation

Transportation planning problems are typically highly complex. It is practical to decompose such problems into a sequence of sub-problems. The four-step planning process is a well-known example. However, this is clearly not ideal, and there are attempts to integrate these four steps as the solution from the four-step process is suboptimal to an integrated model. Continuous network design (NDP), traffic signal setting design (SSD) and traffic assignment are classical problems, and are indeed highly interdependent as well. The route choice decisions in the traffic assignment, which implies the state of the network, are made by travelers based on the link capacities and traffic signal settings. Transportation planners decide on the link capacity improvement and the signal setting design based on the state of the network. If we solve these problems in sequence, we would not have a mutually consistent solution where flow is at user equilibrium, signal settings are optimal, and link capacity improvement decisions are most favorable. Thus, we would prefer an integrated (or combined) model that provides such mutually consistent solutions.

Transportation network improvements consume a substantial portion of the transportation budget. Decisions affecting these improvements could result in significant savings, if they are made using tools that can account for traffic network realities. In particular, the transportation network improvement optimization problems (i.e. NDP, SSD and the integrated NDP, SSD and traffic assignment) are naturally bi-level, and can be viewed as a static version of the non-cooperative, two-person game (Von Stackelberg, 1952) in the context of unbalanced economic markets. With the assumption of perfect information, the static game means that each player has only one move. The leader (transport planners) goes first, attempts to minimize the total costs, and anticipates all possible responses of his opponent, the follower (road users). The follower observes the leader's decision and reacts such that the follower achieves his/her optimal benefit (least

individual's travel time) regardless of external effects (system-wide costs). As such, the bi-level formulations are essential to capture this reality.

Moreover, instead of static traffic assignment models, we employ dynamic traffic assignment (DTA)-based models that propagate traffic according to the cell transmission model (CTM), a traffic flow theoretical model by Daganzo (1994). CTM can capture the traffic interaction among adjacent links. Also, the static traffic assignment models assume steady-state time-invariant O-D demand, leading to suboptimal solutions during the peak period. Furthermore, the transportation network improvement optimization problems rely on several input parameters, especially a long-term O-D demand matrix, which is traditionally treated as deterministic. As estimated future population, demographics and land use patterns are employed to estimate the long-term O-D demand matrix, the O-D demands are indeed random variables. The long-term time-dependent O-D demands should be treated as random variables; then, the stochastic programming techniques are crucial for the resulting stochastic models. The stochastic models can be extended to optimize the tradeoff between the expected cost and the expected risk such that the optimal solutions are the most robust to the O-D demand variation, given the planners' preference on the degree of robustness.

In this dissertation, we formulate several bi-level mathematical models based on the existing DTA linear programs for NDP, SSD and the integrated NDP and SSD under the deterministic, two-stage stochastic programming and robust optimization schemes with the limitation that the models are applicable for only single-destination networks. For larger-size multi-destination networks, diverse metaheuristic algorithms are developed, namely, random search, simulated annealing, genetic algorithm and reactive tabu search.

1.2 Dissertation Contributions

This dissertation can be portrayed as the development of a three-dimensional box filled with proposed mathematical formulations and solution methodologies for transportation network improvement optimization problems as shown in Figure 1.1. The

first dimension depicts the three kinds of bi-level DTA-based transportation network improvement optimization problems: continuous network design problem (NDP), signal setting design (SSD) and integrated design (NDP-SSD). With given available budget, the NDP determines how much investment should be allocated to improve link capacity of pre-specified links in the network. The SSD determines optimal signal settings (cycle lengths, phase sequencing, green splits and offsets) of pre-specified intersections in the network with fixed signal phasing. The NDP-SSD simultaneously solves NDP, SSD and DTA.

The second dimension of the dissertation represents three optimization schemes: deterministic, stochastic and robust. The deterministic optimization considers all input parameters as constant (e.g. mean values or worst-case values) and the objective minimizes the total cost. The objective of stochastic optimization minimizes the expected total cost; i.e. it only considers the first moment of the distribution of total cost. The robust optimization has an objective function that represents a tradeoff between the expected cost and risk. A solution of robust optimization remains close to optimal for all scenarios of input data, whereas a solution of stochastic optimization may yield large changes in the objective value among different scenarios. The first two dimensions constitute nine separate bi-level DTA-based transportation network improvement optimization models: deterministic / stochastic / robust models of NDP / SSD / NDP-SSD.

The third dimension expresses two broad kinds of solution methods: analytical and metaheuristic. Although the analytical approach is limited to solving single-destination and small networks, it is mathematically tractable, enables us to gain insight into the problems, and potentially provides intuition for possible heuristic development. For metaheuristic methods, many assumptions of analytical approach can be relaxed, and large-scale problems can be solved with the use of any existing simulation-based DTA software. Four types of metaheuristics are developed, namely, random search, simulated annealing, reactive tabu search and genetic algorithm. The first three metaheuristics are

the single solution-based algorithms, whereas the last one is the population-based algorithm.

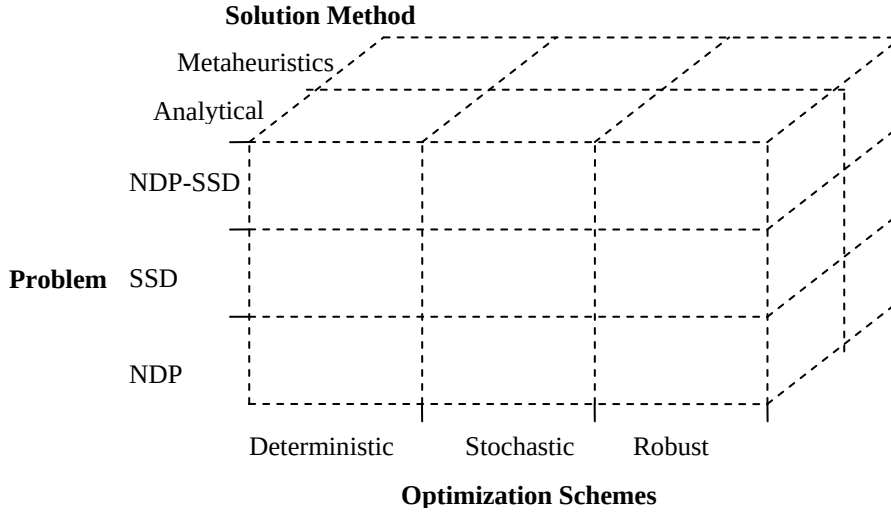


Figure 1.1. Dissertation Framework

Specifically, in our view, the contributions of this dissertation include:

1. Three new formulations of dynamic continuous network design problems:
 - Deterministic linear bi-level program (BLPNDP),
 - Two-stage stochastic linear bi-level program (SLP2-BLPNDP), and
 - Robust bi-level model (RO-BLPNDP)
2. Six new mathematical formulations of the combined DTA and SSD:
 - Deterministic user-optimal model (UOSSD),
 - Two-stage stochastic system-optimal model (SLP2-SOSSD),
 - Two-stage stochastic user-optimal model (SLP2-UOSSD),
 - Deterministic bi-level model (BLPSSD),
 - Two-stage stochastic bi-level model (SLP2-BLPSSD), and
 - Robust bi-level model (RO-BLPSSD)
3. Six new formulations of the integrated NDP, SSD and DTA problem:
 - Deterministic user-optimal model (UONDP-SSD)
 - Deterministic bi-level model (BLPNDP-SSD)

- Two-stage stochastic system-optimal model (SLP2-SONDP-SSD)
 - Two-stage stochastic user-optimal model (SLP2-UONDP-SSD)
 - Two-stage stochastic bi-level model (SLP2-BLPNDP-SSD)
 - Robust bi-level model (RO-BLPNDP-SSD)
4. Two exact solution methods for NDP
 - Modified K^{th} best algorithm
 - Karush-Kuhn-Tucker based mixed integer programming reformulation
 5. Mathematical proofs used in the exact solution methods for SSD and NDP-SSD
 - The reduction of a mixed-zero-one continuous linear bi-level program to an equivalent parametric linear bi-level program
 - The reduction of a mixed-zero-one continuous quadratic-linear bi-level program to an equivalent parametric quadratic-linear bi-level program
 6. Metaheuristic Algorithmic Development
 - Pure random search
 - Simulated annealing
 - Modified simulated annealing
 - Genetic algorithm
 - Reactive tabu search based on three different neighborhood definitions
 7. Comparison of the new NDP, SSD and NDP-SSD models with the existing models using small hypothetical test networks to illustrate the worthiness of the proposed robust optimization models and to demonstrate the benefit of the integrated models over the sequential approach.
 8. Comparison of metaheuristics performance for the nine problems (deterministic/ stochastic/ robust optimization models of NDP/ SSD/ NDP-SSD) on two limited multi-origin multi-destination networks to identify the winners on all nine problems.

1.3 Dissertation Structure

A critical review of the relevant literature is described in Chapter 2. The review on the transportation network improvement optimization problems includes traffic assignment-based network design problem, traffic assignment-based signal setting design problem and combined network design, signal setting design and traffic assignment problem. Also, the relevant dynamic traffic assignment models, which are based on the cell transmission model by Daganzo (1994), are described. The review on the optimization techniques includes stochastic programming, robust optimization, linear bi-level programming, zero-one continuous linear bi-level programming and metaheuristics.

In Chapter 3, the existing single-level linear programming models of DTA-based NDP are described, and three new bi-level DTA-based NDP models under deterministic, stochastic and robust optimization frameworks are proposed. Then, six new mathematical programming models of DTA-based SSD, which are extended from the known single-level mixed integer programming model of system-optimal SSD problem, are described. Similarly, six new models of DTA-based NDP-SSD problem are proposed, based on the existing single-level system-optimal DTA-based NDP-SSD model. This chapter concludes with the exact solution methods for all analytical models.

Chapter 4 describes the decision variables of the considered three transportation network improvement optimization problems and the associated solution representations and decoding procedures for the four metaheuristic methods. The simulation-based objective functional evaluation procedures for deterministic, stochastic and robust optimization schemes are discussed. Then, it describes the developed metaheuristics including simulated annealing, modified simulated annealing, genetic algorithm, reactive tabu search and random search.

Chapter 5 presents the computational experiences of analytical approach on small single-destination cell transmission networks, and Chapter 6 shows the experimental results of the simulation-based implementation of the four metaheuristics on limited multi-destination test networks. Finally, Chapter 7 discusses possible extensions, and summarizes the major findings.

Chapter 2. Literature Review

This chapter presents a review of the literature to provide a basis for understanding the models and algorithms proposed in this dissertation. It starts off by reviewing network design problem (NDP), followed by the combined traffic assignment and signal setting design problem (SSD) and the integrated traffic assignment, NDP and SSD problem (NDP-SSD). Then, dynamic traffic assignment (DTA) is discussed, and five optimization schemes are described, namely, stochastic optimization, robust optimization, linear bi-level optimization, zero-one continuous linear bi-level programming and metaheuristics.

2.1 Network Design Problem

The literature review section on network design problem (NDP) is limited to only a few models, which are representative of the methodologies used to approach the problem, and it is by no means comprehensive. The comprehensive reviews of traffic assignment based transportation NDP can be found in Magnanti and Wong (1984), Boyce (1984), Friesz (1985), and Yang and Bell (1998). Typically, the NDP objective function minimizes the total system travel time (TSTT) subject to the conservation of flow conditions and budget constraint, or minimizes the TSTT plus the cost converted to equivalent time unit subject to the conservation of flow conditions. The NDP formulations can be classified according to the following four criteria: 1) system-optimal (SO) / user-optimal (UO) behavior, 2) static / dynamic traffic assignment, 3) discrete / continuous investment variable and 4) deterministic / stochastic parameters. The SO behavior is mathematically tractable but unrealistic, whereas the UO behavior is typically complicating the problem but more realistic. The static traffic assignment assumes the steady-state condition, whereas the dynamic traffic assignment (DTA) accounts for time dynamics. The discrete investment variable allows entire-lane or new link addition, whereas the continuous investment permits a fraction of lane addition. The continuous investment variable has extensively been employed in the literature, and the justification

is that because most roads in the urban area are already constructed, discrete variables are not practical. The continuous link expansion can be implemented by altering lane width, median and shoulder area. Alternatively, the continuous NDP can be considered as a possible heuristic for the discrete NDP. Lastly, the problem parameters have typically been considered deterministic as opposed to stochastic.

The literature on the NDP based on SO static traffic assignment and deterministic parameters is such as LeBlanc (1975), Hoang (1982), LeBlanc and Abdulaal (1979), Dantzig et al. (1979), and LeBlanc and Abdulaal (1984). LeBlanc (1975) formulated the problem as a mixed integer nonlinear programming (MINLP) model with discrete investment variables, and suggested a branch-and-bound algorithm. Hoang (1982) proposed the generalized Benders' partition method to solve the MINLP model. LeBlanc and Abdulaal (1979) proposed a computationally efficient dual approach to the SO NDP with continuous investment variables. Dantzig et al. (1979) formulated the problem as a convex NLP model allowing continuous and discrete link expansions, and employed a decomposition algorithm. LeBlanc and Abdulaal (1984) compared the approximate representations of the continuous UO and SO NDP models, and showed that the solutions from the SO NDP model were as good as those from the UO NDP.

The literature on the NDP based on UO static traffic assignment and deterministic parameters is such as Abdulaal and LeBlanc (1979), Suwansirikul et al. (1987), Marcotte (1983), Meng et al. (2001), Friesz et al. (1992), LeBlanc and Boyce (1986), Patriksson and Rockafellar (2002), and Mouskos (1991). Abdulaal and LeBlanc (1979) formulated the continuous NDP as an unconstrained NLP model, and showed that Hooke-Jeeves algorithm in Hooke and Jeeves (1961) outperformed the Powell's method in Powell (1964). Suwansirikul et al. (1987) showed that a heuristic, Equilibrium Decomposed Optimization, was more computationally efficient than Hooke-Jeeves algorithm. Marcotte (1983) suggested a constraint accumulation algorithm and the iterative optimization assignment (IOA), which is efficient but may not converge to an optimal solution. Meng et al. (2001) provided an equivalent continuously differentiable non-convex model, and found an exact local solution by applying an efficient convergent

augmented Lagrangian method. Friesz et al. (1992) proposed a simulated annealing approach to solve the continuous non-convex NLP model with variational inequality constraints. This method is computationally expensive, but yields near-global optimal solutions. LeBlanc and Boyce (1986) re-formulated the continuous model as a bi-level linear program, and suggested the efficient point algorithm by Bard (1983) coupled with the Frank-Wolfe approximation for large network. Later, Marcotte (1988) proved that Bard's efficient point algorithm is unable to find the true optimum by a counter example. However, the bi-level program is still valid, and other existing algorithms that guarantee the exact solution may be applicable. Patriksson and Rockafellar (2002) showed that a constrained, locally Lipschitz minimization problem is equivalent to the mathematical program with equilibrium constraints (MPEC), and suggested a descent algorithm. Mouskos (1991) proposed a tabu-search based algorithm to solve the discrete network design problem.

The stochastic user optimal (SUO) assignment is based on the assumption that an individual's perceived travel costs are subject to random error. The literature on the NDP based on SUO assignment is such as Davis (1994) and Chen and Alfa (1991). Davis (1994) showed that the continuous NDP with the logit-based SUO assignment leads to a differentiable, large-scale, and tractable problem, and suggested a procedure for computing the derivatives of the SUO assignment without computing the route choice probabilities. Later, Yang and Bell (1998) pointed out that this model may yield some unreasonable results because the model generally overestimates traffic flow on overlapping routes. Chen and Alfa (1991) solved the discrete model by a heuristic based on the branch-and-bound method and the logit-based incremental traffic assignment algorithm.

The above-mentioned static NDP models have three drawbacks when compared with the DTA-based models. First, the static models cannot capture the traffic interaction among adjacent links, which could lead to erroneous solutions such as bottlenecks. In contrast, the DTA-based model captures traffic propagation among adjacent links, thus avoiding potential bottlenecks. Second, the static models assume steady-state time-

invariant OD demand; this is obviously unrealistic during the peak period and leads to suboptimal solutions. On the contrary, the DTA-based models overcome this deficiency by capturing traffic dynamics and by providing recommendations that optimize the network for the duration of the assignment. Third, in the static models, the capacity expansions typically appear in the denominator of the link performance function, which creates nonlinearities that are difficult to handle in a mathematical program. In the DTA-based model, the capacity expansions for each road segment can be captured by a single parameter on the right-hand side of a linear constraint.

The literature on the DTA-based NDP is very limited, such as Janson (1995), Waller et al. (2006), Li et al. (2000), Ukkusuri and Waller (2004), Waller and Ziliaskopoulos (2001), Ukkusuri et al. (2004), Jeon et al. (2005) and Karoonsoontawong and Waller (2005). Janson (1995) showed that the NDP solution based on static UO assignment might not reduce the TSTT when evaluating the solution by the dynamic UO assignment. Waller et al. (2006) proposed a continuous NDP formulated as an LP model where the users' route choices are based on the single-destination SO DTA LP model with fixed departure time O-D demands introduced by Ziliaskopoulos (2000). The SO DTA LP model propagates traffic according to the cell transmission traffic flow theoretical model by Daganzo (1994,1995). The extension of the SO DTA based NDP LP model included the multiple destination with fixed arrival time O-D demands as discussed in Waller (2000). Li et al. (2000) suggested the Dantzig-Wolfe decomposition procedure in which the subproblem is the minimum cost flow problem, and showed that it yielded a great reduction in computational time. Ukkusuri and Waller (2004) proposed a continuous NDP formulated as an LP model where the users' route choices are based on the UO DTA LP model by Waller and Ukkusuri (2003) also employing Daganzo's cell transmission model. Further, Waller and Ziliaskopoulos (2001) introduced the stochastic SO DTA-based NDP with long term single-destination O-D demand uncertainty, formulated as a two-stage stochastic linear program (SLP2) with recourse and a chance-constrained program (CCP). They tested the individual CCP on a test problem, and concluded that planning for an inflated demand might produce benefits in

terms of system performance and reduced variance. Waller (2000) solved the SLP2 model by the simplex method; this was possible only because of the considered small number of demand realizations. Ukkusuri et al. (2004) formulated the stochastic UO DTA-based NDP with long term single-destination O-D demand uncertainty as an SLP2 model with recourse and a CCP, and only solved the CCP with a similar conclusion to Waller and Ziliaskopoulos (2001). Jeon et al. (2005) employed a genetic algorithm to solve the discrete NDP by allowing only one lane addition. Karoonsoontawong and Waller (2005) comprehensively compared the SO and UO versions of SLP2 models using Monte Carlo bounding techniques.

In this dissertation, we complement the literature on the continuous DTA-based NDP by proposing three new bi-level NDP models in Chapter 3: deterministic, two-stage stochastic and robust.

2.2 Traffic Assignment-Based Signal Optimization

A comprehensive review of the combined signal optimization and traffic assignment problem can be found in Lee (1998). This literature review section is limited to certain models, which are representative of the methodologies used to approach the problem. The literature on traffic assignment based signal optimization can be classified according to the following criteria: 1) system optimal / user equilibrium behavior and 2) static / dynamic traffic assignment.

The literature on the combined signal optimization and static user equilibrium problem is such as Allsop (1974), Allsop and Charlesworth (1977), Charlesworth (1977), Gartner (1974, 1976), Lee (1998), Sun and Mouskos (2000), and Teklu (2003). Allsop (1974) and Gartner (1974, 1976) were the first who independently suggested the necessity of this problem. That is, the routing of traffic in a network according to Wardrop's first principle depends on signal settings and should ideally be considered simultaneously with signal setting design. Allsop (1974) suggested a conceptual algorithm of the iterative optimization assignment (IOA) procedure, which iteratively solves two subproblems: signal optimization and static traffic assignment. Allsop and

Charlesworth (1977) and Charlesworth (1977) extended the conceptual algorithm of Allsop (1974). Lee (1998) compared various local search algorithms and developed a simulated annealing algorithm and a genetic algorithm. Sun and Mouskos (2000) formulated the problem as a bi-level nonlinear mixed integer programming model. Teklu (2003) applied a genetic algorithm to solve the problem.

The literature on the combined signal optimization and SO DTA problem can be found in Abdelfatah and Mahmassani (1998), Varia and Dhigra (2004), and Waller (2000). The SO DTA is based on Wardrop's second principle: "drivers cooperate with one another in order to minimize total system time" (Wardrop, 1952). In Abdelfatah and Mahmassani (1998)'s formulation, a central controller has complete a priori information on time-varying O-D desires for the entire assignment duration. The controller assigns a path to the desired destination to each vehicle as it enters the network and optimizes the signal settings to optimize certain indicator of the system performance subject to the applicable constraints. Varia and Dhingra (2004) employed a genetic algorithm. Waller (2000) formulated the problem as a mixed-integer program based on Ziliaskopoulos (2000)'s SO DTA LP model. Our analytical work in Chapter 3 extends his work by formulating the problem as bi-level programs.

The literature on the combined signal optimization and dynamic user equilibrium problem is such as Smith and Ghali (1990), Chen and Hsueh (1997), and Gartner and Stamatiadis (1997). Smith and Ghali (1990) considered the problem in an increasingly dynamic way; however, they were unable to obtain results when dynamic demand and responsive signal control are combined. Chen and Hsueh (1997) proposed a bi-level framework, and proposed a heuristic IOA algorithm. Gartner and Stamatiadis (1997) presented a conceptual framework for integrating DTA and a real-time traffic adaptive control system. Neither solution algorithm nor illustrative example was shown in the paper.

In terms of signal optimization, five variables are cycle lengths, green splits, time offset, phase sequencing and signal phasing. The work that optimizes only green splits and assumes other variables as fixed values is such as Lee (1998), Tan et al. (1979),

Smith and Van Vuren (1993) and Cipriani and Fusco (2004). Those that consider only common cycle length and green splits are such as Abdelfatah and Mahmassani (1998) and Varia and Dhingra (2004). When time offsets are assumed fixed, it implies that the benefit of signal coordination is not exploited. Those considering common cycle length, green splits and time offset are such as Wong and Yang (1999), Sun and Mouskos (2000), Teklu (2003) and Gartner et al. (1980). In this dissertation, we optimize cycle lengths, green splits, time offsets and phase sequencing with fixed phase ordering. Further, we allow intersections to have different cycle times, so our models are less constrained than those assuming a common cycle length.

2.3 Combined Network Design, Signal Optimization and Traffic Assignment

The literature on the combined signal optimization, continuous network design and static user equilibrium can be found in Ziyou and Yifan (2002) and Chiou (2005). Ziyou and Yifan (2002) formulated the problem as a bi-level program, and proposed a heuristic solution algorithm based on sensitivity analysis. They employed a link cost function as a function of traffic flows, signal timing variables, link capacity increases and the O-D matrix multipliers, and optimized only cycle time and green splits. Chiou (2005) formulated a mathematical program with equilibrium constraints, and proposed a projected Quasi-Newton heuristic. He optimized a common cycle time, green splits and offsets.

The literature on the combined signal optimization, continuous network design and SO DTA is such as Waller (2000). He formulated a mixed-integer program based on his SO NDP LP model. Our analytical work in Chapter 3 extends his work by combining continuous network design, signal optimization and UO DTA, and takes into account the O-D demand uncertainty.

2.4 Dynamic Traffic Assignment

The traffic assignment problem determines the driver path assignment and the corresponding path costs with the assumption of driver behavior (Sheffi, 1985). The

assumptions for path choice behavior are Wardrop's (1952) principles of system optimum (SO) and user equilibrium (UE) flow distribution. For SO traffic assignment, all drivers select paths such that the total system travel cost is minimized. For UE traffic assignment, all drivers select paths such that their individual travel costs cannot be improved by a unilateral decision to change paths. The dynamic traffic assignment (DTA) problem determines time-dependent path assignment and corresponding time-dependent path costs. It is noted that Ran and Boyce (1996) employed two different terms: dynamic user optimal (DUO) and dynamic user equilibrium (DUE). In this dissertation, we use these two terms interchangeably, and assume that every driver has perfect foresight and selects the shortest path based on the actually experienced path costs. Chang (2004) classified DTA models into two broad categories: analytical and simulation-based approaches. The analytical approach includes mathematical programming, variational inequality and control theory. An overview and discussion of DTA can be found in Peeta and Ziliaskopoulos (2001). We will describe relevant DTA LP models and a simulation-based DTA model as well as cell transmission model in which the relevant DTA models are based upon.

2.4.1 Cell Transmission Model

Daganzo (1994) transformed the differential equations of the hydrodynamic model proposed by Lighthill and Whitham (1955) and Richards (1956) into simple difference equations with the assumption that the relationship between flow and density at the cell level is piecewise linear. The resulting model was named the cell transmission model (CTM). The CTM can represent the non-linear relationship between density and traffic flow at the link level although it assumes the piecewise linear relationship at the cell level. The CTM can accurately portray traffic propagation on street networks and capture traffic phenomena such as disturbance propagation and creation of shockwaves on freeways. It can easily be modified to account for traffic signal control and ramp metering device. Additionally, one can adjust the computational requirements by considering the tradeoff between the discretization interval and the required accuracy.

Daganzo (1994) considered the piecewise-linear relationship between traffic flow (q) and density (k) as follows:

$$q = \min\{vk, q_{\max}, w(k_j - k)\}, \text{ for } 0 \leq k \leq k_j \quad (2.1)$$

where v is free flow speed, $q_{\max}$ is maximum flow or capacity, w is backward propagation speed, and k_j is jam density. As such, the equations by Lighthill and Whitham (1955) and Richards (1956) for a single highway link can be approximated by a set of difference equations, and the state of the system is updated at every time interval. The CTM model discretizes the assignment period into small time intervals, and divides every link of the street network into small homogeneous segments (cells) such that the cell length is equal to the distance traveled by free-flow speed in a time interval.

Let us denote C as the set of cells, which are categorized into five subsets as shown in Figure 2.1: ordinary cells (C_O), diverging cells (C_D), merging cells (C_M), source cells (C_R) and sink cells (C_S). E is the set of cell connectors, which are divided into five subsets: ordinary cell connectors (E_O), diverging cells (E_D), merging cell connectors (E_M), source cell connectors (E_R) and sink cell connectors (E_S). A cell connector connects a cell to another. Ziliaskopoulos (2000) employed the term “cell connector” rather than “link” used in Daganzo (1994) to avoid confusion with the links of the street network. The cell connectors do not hold any traffic, but simply designate the way vehicles are transferred from one cell to another. T is the set of discrete time intervals: $T = \{\tau, 2\tau, \dots, |T|\tau\}$. τ can be assumed 1 without loss of generality. The state of the system at any time instant t is the number of vehicles contained in each cell, $x_i^t, i \in C, t \in T$. The flow moving from cells i to j at time interval t is denoted by $y_{ij}^t, (i, j) \in E, t \in T$. For each cell, the following three constants are defined: N_i^t = maximum number of vehicles that can be present in cell i at time interval t (equal infinity for source and sink cells); Q_i^t = maximum number of vehicles that can flow into or out of cell i during time interval t (equal infinity for sink cell); and δ_i^t = ratio v/w for cell i and time interval t . These constants can vary with time to capture time-dependent capacity and flow variations. Daganzo (1994) showed that these constants and the length of the cell can reasonably

capture the road characteristics. In addition, the set of predecessor cells to cell i is denoted by $BS(i)$, and the set of successor cells to cell i denoted by $FS(i)$.

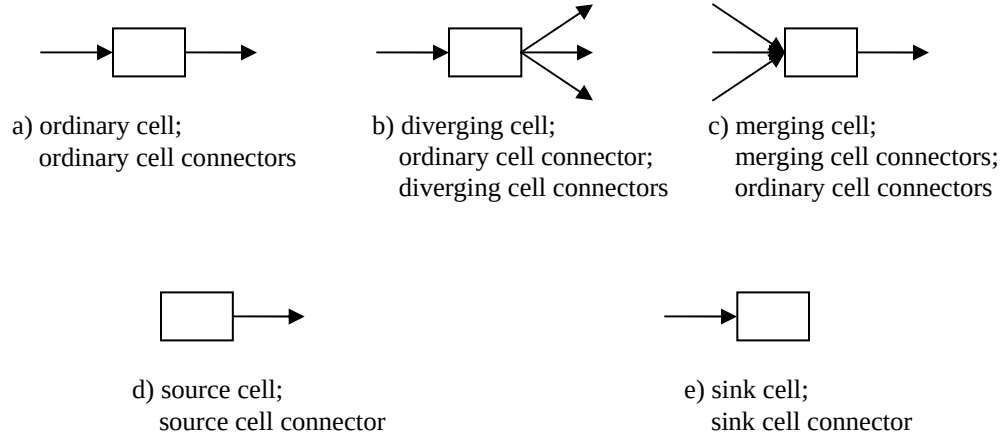


Figure 2.1. Cells and Cell Connectors (Ziliaskopoulos, 2000)

The basic relationships of the CTM that describe the evolution of traffic flow are expressed as follows (Ziliaskopoulos, 2000).

Ordinary Cells and Ordinary Cell Connectors

Equation 2.2 is the cell mass conservation for ordinary cells, and Equation 2.3 states that the flow between two ordinary cells is bound by four quantities: the number of vehicles in the beginning cell, the remaining capacity at the ending cell, the maximum flows that can get out of the beginning cell and into the ending cell.

$$x_i^t = x_i^{t-1} + y_{ki}^{t-1} - y_{ij}^{t-1}, \quad k \in BS(i), j \in FS(i), \forall i \in C_o, \forall t \in T \quad (2.2)$$

$$y_{ki}^t = \min\{x_k^t, \min[Q_i^t, Q_k^t], \delta_i^t (N_i^t - x_i^t)\}, \quad \forall (k, i) \in E_o, \forall t \in T \quad (2.3)$$

Diverging Cells and Diverging Cell Connectors

Equation 2.4 is the cell mass conservation for diverging cells:

$$x_i^t = x_i^{t-1} + y_{ki}^{t-1} - \sum_{j \in FS(i)} y_{ij}^{t-1}, \quad k \in BS(i), \forall i \in C_D, \forall t \in T \quad (2.4)$$

As shown in Figure 2.1, the incoming cell connectors of diverging cell is an ordinary cell, so y_{ki}^t is determined by Equation 2.3. The outflows y_{ij}^t are exogenously determined according to Daganzo (1994); however, Ziliaskopoulos suggested a linear program for generating the turning proportions.

Merging Cells and Merging Cell Connectors

Equation 2.5 is the cell mass conservation for merging cells:

$$x_i^t = x_i^{t-1} + \sum_{k \in BS(i)} y_{ki}^{t-1} - y_{ij}^{t-1}, \quad j \in FS(i), \forall i \in C_M, \forall t \in T \quad (2.5)$$

The outgoing cell connector of merging cell is an ordinary cell connector, so the outflow y_{ij}^t is determined by Equation 2.3. The inflows y_{ki}^t can be determined by the linear program suggested by Ziliaskopoulos (2000). Daganzo (1994) proposed that signal control and priorities at intersections can be modeled by defining a time-dependent binary parameter p_{ki}^t such that $\sum_{\forall k \in BS(i)} p_{ki}^t = 1$. Then, the inflows y_{ki}^t of merging cells are

determined by Equation 2.6:

$$y_{ki}^t = \min\{x_k^t, Q_k^t, p_{ki}^t Q_i^t, \delta_i^t (N_i^t - x_i^t)\}, \quad \forall (k, i) \in E_M, \forall t \in T \quad (2.6)$$

This concept was employed in the analytical work for signal optimization in Waller (2000) as well as in this dissertation where p_{ki}^t becomes variable.

Source Cells and Source Cell Connectors

The source cells have infinite capacity ($N_i^t = \infty, \forall i \in C_R, \forall t \in T$) but finite output flow. The cell mass conservation of the source cell is:

$$x_i^t = x_i^{t-1} + d_i^t - y_{ij}^{t-1}, \quad j \in FS(i), \forall i \in C_R, \forall t \in T \quad (2.7)$$

where d_i^t is the demand or inflow in cell i in time interval t . The initial values x_i^0 can be set to initial traffic conditions in the network at the beginning of the simulation period. The outflow y_{ij}^t for a source cell connector is determined by Equation 2.3.

Sink Cell and Sink Cell Connector

Since the sink cell is allowed to have infinite capacities ($N_i^t = \infty, \forall i \in C_s, \forall t \in T$) and infinite inflows ($Q_i^t = \infty, \forall i \in C_s, \forall t \in T$), their occupancies are not constrained. The inflow to a sink cell is determined by the state and characteristics of its predecessor cell.

2.4.2 Dynamic Traffic Assignment Linear Programming Models

Ziliaskopoulos (2000) introduced a linear programming (LP) formulation for the single-destination SO DTA problem based on Daganzo's CTM for traffic propagation. Ukkusuri (2002) introduced a UO DTA LP formulation based on the CTM for single-destination problem. These two models circumvent the need for link performance functions as the flow propagates according to the CTM, thus capturing more traffic realities. The DTA LP models use cells to represent link flow, so it can capture flow variability inside the link, which is not easily possible if traffic is propagated by using link exit functions. The two models are limited to a single destination and do not overcome the general drawbacks of the analytical models. However, they can capture realities of actual networks reasonably well, since their underlying structure is based on a traffic flow theoretical model. The first-in-first-out condition is satisfied by the models because there is no incentive for the LP to favor one vehicle over another in the same cell provided that both vehicles are going to the same destination. Although these are not operational models for real-world applications, they provide some insights on the DTA problem properties, and enable the analytical studies of the bi-level NDP, the bi-level SSD, and the bi-level NDP-SSD considered in this dissertation. The next sub-section describes the SO DTA LP model, followed by the UO DTA LP model and the concise formulations employed in this dissertation.

2.4.2.1 System Optimal Dynamic Traffic Assignment

Ziliaskopoulos (2000) formulated the SO DTA as an LP model. The objective is to minimize the total system travel time (TSTT) in the network; i.e. the travel time experienced by all users of the network. At any time interval t , the travel time

experienced by the users of cell i equals to $\tau \cdot x_i^t$ because according to the cell transmission model, these users have to stay in this cell for the duration of the time interval. The travel time experienced by all users of the network during time interval t is

$\sum_{\forall i \in C \setminus C_s} \tau \cdot x_i^t$, since no users are stored at the cell connectors, and the sink cell does not

contribute to the TSTT. Thus, the TSTT during the whole assignment period T is

$\sum_{\forall t \in T} \sum_{\forall i \in C \setminus C_s} \tau \cdot x_i^t$, a linear relationship of x_i^t . Ukkusuri (2002) expressed the TSTT as the

difference of the arrival and departure times for every unit of flow within the network:

$$\text{TSTT} = \sum_{\forall k} A_k - \sum_{\forall k} D_k \quad (2.8)$$

where A_k is the arrival time of some unit of flow k at the destination, and D_k is the departure time of the same unit of flow from its origin. Since this approach is departure-time based, the departure time of each demand is known. Thus, $\sum_{\forall k} D_k$ is a constant and

equals to $\sum_{t \in T} \sum_{i \in C_R} t \cdot d_i^t$ where d_i^t is the demand at cell i in time t . Since the departure

times are constant, these can be dropped from the objective function without changing the optimal solution. Hence, an equivalent objective is to minimize the summation of arrival times. To find the arrival times, we only examine the flow leaving the network, and assign a cost on this flow equal to the time interval that it departs, i.e.

$\sum_{\forall k} A_k = \sum_{t \in T} \sum_{\forall (i,j) \in E_s} \tau \cdot t \cdot y_{ij}^t$. Thus, the objective function of SO DTA is

$$\min \sum_{\forall i \in C \setminus C_s} \tau \cdot x_i^t = \min \sum_{\forall t \in T} \sum_{\forall (i,j) \in E_s} \tau \cdot t \cdot y_{ij}^t.$$

Since τ can take any positive value, the equivalent objective without affecting the solution of the LP is

$$\min \sum_{\forall i \in C \setminus C_s} x_i^t = \min \sum_{t \in T} \sum_{\forall (i,j) \in E_s} t \cdot y_{ij}^t.$$

It is noted that we refer to TSTT as $\sum_{t \in T} \sum_{\forall (i,j) \in E_s} t \cdot y_{ij}^t$ for our analytical work in Chapters 3

and 5. The LP formulation of the single-destination SO DTA is (Ziliaskopoulos, 2000):

$$\min_{x,y} \sum_{\forall t \in T} \sum_{\forall (i,j) \in E_s} t \cdot y_{ij}^t \quad (2.9.1a)$$

subject to

$$x_i^t - x_i^{t-1} - \sum_{k \in BS(i)} y_{ki}^{t-1} + \sum_{j \in FS(i)} y_{ij}^{t-1} = d_i^t, \quad \forall i \in C \setminus C_s, \forall t \in T \quad : \lambda_i^t \quad (2.9.2)$$

$$y_{ij}^t - x_i^t \leq 0, \quad \forall (i,j) \in E_o \cup E_R \cup E_S \cup E_M, \forall t \in T \quad : \mu_{ij}^{1t} \quad (2.9.3)$$

$$\sum_{\forall j \in FS(i)} y_{ij}^t - x_i^t \leq 0, \quad \forall i \in C_D, \forall t \in T \quad : \pi_i^{1t} \quad (2.9.4)$$

$$y_{ij}^t \leq Q_j^t, \quad \forall (i,j) \in E_o \cup E_D \cup E_R, \forall t \in T \quad : \mu_{ij}^{2t} \quad (2.9.5)$$

$$\sum_{\forall k \in BS(i)} y_{ki}^t \leq Q_i^t, \quad \forall i \in C_M, \forall t \in T \quad : \pi_i^{2t} \quad (2.9.6)$$

$$y_{ij}^t \leq Q_i^t, \quad \forall (i,j) \in E_o \cup E_R \cup E_S \cup E_M, \forall t \in T \quad : \mu_{ij}^{3t} \quad (2.9.7)$$

$$\sum_{\forall j \in FS(i)} y_{ij}^t \leq Q_i^t, \quad \forall i \in C_D, \forall t \in T \quad : \pi_i^{3t} \quad (2.9.8)$$

$$y_{ij}^t \leq \delta_j^t (N_j^t - x_j^t), \quad \forall (i,j) \in E_o \cup E_D \cup E_R, \forall t \in T \quad : \mu_{ij}^{4t} \quad (2.9.9)$$

$$\sum_{\forall k \in BS(i)} y_{ki}^t \leq \delta_i^t (N_i^t - x_i^t), \quad \forall i \in C_M, \forall t \in T \quad : \pi_i^{4t} \quad (2.9.10)$$

$$x_i^0 = \varsigma_i \quad \forall i \in C \setminus C_s \quad : \lambda_i^0 \quad (2.9.11)$$

$$y_{ij}^0 = 0 \quad \forall (i,j) \in E \quad : \theta_{ij} \quad (2.9.12)$$

$$x_i^{[T]} = 0, \quad \forall i \in C \setminus C_s \quad : \tau_i \quad (2.9.13)$$

$$x_i^t \geq 0, \quad \forall i \in C \setminus C_s, \forall t \in T \quad (2.9.14)$$

$$y_{ij}^t \geq 0, \quad \forall (i,j) \in E, \forall t \in T \quad (2.9.15)$$

Constraint 2.9.2 is a concise form of the cell mass conservation expressions (Equations 2.2, 2.4, 2.5 and 2.7). Constraints 2.9.3-2.9.10 are a more relaxed statement of Equation 2.3. These constraints imply that although the capacity and flow constraints are not violated, the whole allowable traffic will not necessarily exit cell. This allows the model

to account for traffic holding for SO DTA if this leads to an improved SO solution. Constraints 2.9.11 and 2.9.12 specify the initial flow conditions. Demands (d_i^t) and initial occupancies (ς_i) are given. In this dissertation, ς_i are always set to zero. Constraints 2.9.14-2.9.15 are non-negativity constraints. Constraint 2.9.13 enforces that all vehicles exit the network before the last time interval. Without this constraint, an erroneous solution may result; i.e., the vehicles that do not have sufficient time to exit the network may be held in a cell, because moving them would not make a difference to the objective function. It is noted that the corresponding dual variables are shown at the end of each constraint.

The dual program of SO DTA is:

$$\text{Max} \sum_{\forall t \in T} \left(\sum_{\forall i \in C_R} d_i^t \lambda_i^t + \sum_{\forall (i,j) \in E_O \cup E_D \cup E_R} Q_j^t \mu_{ij}^{2t} + \sum_{\forall i \in C_M} Q_i^t \pi_i^{2t} + \sum_{\forall i \in C_R} Q_i^t \pi_i^{3t} \right) + \sum_{\forall i \in C \setminus C_S} \varsigma_i \lambda_i^0 \quad (2.10.1)$$

Subject to

$$\lambda_i^t - \lambda_i^{t+1} - \mu_{ij}^{1t} + \delta_i^t \mu_{ki}^{4t} + \eta_0^t \tau_i \leq 0, \quad \forall i \in C_O, \forall t \in T \quad : x_i^t \quad (2.10.2)$$

$$\lambda_i^t - \lambda_i^{t+1} - \pi_i^{1t} + \delta_i^t \mu_{ki}^{4t} + \eta_0^t \tau_i \leq 0, \quad \forall i \in C_D, \forall t \in T \quad : x_i^t \quad (2.10.3)$$

$$\lambda_i^t - \lambda_i^{t+1} - \mu_{ij}^{1t} + \delta_i^t \pi_i^{4t} + \eta_0^t \tau_i \leq 0, \quad \forall i \in C_M, \forall t \in T \quad : x_i^t \quad (2.10.4)$$

$$\lambda_i^t - \lambda_i^{t+1} - \mu_{ij}^{1t} + \eta_0^t \tau_i \leq 0, \quad \forall i \in C_R, \forall t \in T \quad : x_i^t \quad (2.10.5)$$

$$\lambda_i^0 - \lambda_i^1 \leq 0, \quad \forall i \in C \setminus C_S \quad : x_i^0 \quad (2.10.6)$$

$$\lambda_i^{t+1} - \lambda_j^{t+1} + \mu_{ij}^{1t} + \mu_{ij}^{2t} + \mu_{ij}^{3t} + \mu_{ij}^{4t} \leq 0, \quad \forall (i, j) \in E_O \cup E_R, t \in T \quad : y_{ij}^t \quad (2.10.7)$$

$$\lambda_i^{t+1} - \lambda_j^{t+1} + \pi_i^{1t} + \mu_{ij}^{2t} + \pi_i^{3t} + \mu_{ij}^{4t} \leq 0, \quad \forall (i, j) \in E_D, \forall t \in T \quad : y_{ij}^t \quad (2.10.8)$$

$$\lambda_i^{t+1} - \lambda_j^{t+1} + \mu_{ij}^{1t} + \pi_j^{2t} + \mu_{ij}^{3t} + \pi_j^{4t} \leq 0, \quad \forall (i, j) \in E_M, t \in T \quad : y_{ij}^t \quad (2.10.9)$$

$$\lambda_i^{t+1} + \mu_{ij}^{1t} + \mu_{ij}^{3t} \leq t, \quad \forall (i, j) \in E_S, \forall t \in T \quad : y_{ij}^t \quad (2.10.10a)$$

$$\theta_{ij} + \lambda_i^1 - \lambda_j^1 \leq 0, \quad \forall (i, j) \in E \setminus E_S \quad : y_{ij}^0 \quad (2.10.11)$$

$$\theta_{ij} + \lambda_i^1 \leq 0, \quad \forall (i, j) \in E_S \quad : y_{ij}^0 \quad (2.10.12)$$

$$\lambda_i^t, \forall i \in C \setminus C_S, \forall t \in T \text{ unrestricted in sign} \quad (2.10.13)$$

$$\theta_{ij}, \forall (i, j) \in E \text{ unrestricted in sign} \quad (2.10.14)$$

$$\lambda_i^0, \tau_i, \forall i \in C \setminus C_S \text{ unrestricted in sign} \quad (2.10.15)$$

$$\pi_i^{1t}, \pi_i^{3t} \leq 0, \forall i \in C_D, \forall t \in T \quad (2.10.16)$$

$$\pi_i^{2t}, \pi_i^{4t} \leq 0 \quad \forall i \in C_M, \forall t \in T \quad (2.10.17)$$

$$\mu_{ij}^{1t}, \mu_{ij}^{3t} \leq 0, \forall (i, j) \in E_O \cup E_R \cup E_S \cup E_M, \forall t \in T \quad (2.10.18)$$

$$\mu_{ij}^{2t}, \mu_{ij}^{4t} \leq 0, \forall (i, j) \in E_O \cup E_D \cup E_R, \forall t \in T \quad (2.10.19)$$

where η_0^t equals 0 if $t \in T \setminus \{T\}$, and equals 1, otherwise. The interpretation of dual variables and the proof of a necessary and sufficient condition for SO DTA on a single destination network are referred to Ziliaskopoulos (2000).

2.4.2.2 User Optimal Dynamic Traffic Assignment

Ukkusuri (2002) formulated the UO DTA as an LP, which is directly based on the SO DTA LP model. He argued that by changing the cost differential in the objective function from uniform to an exponentially decreasing amount, user optimal conditions can be realized. This vector of costs is denoted as M_t . For user optimal conditions, M_t must be defined in such a way that the objective values earlier arrivals to such an extent that it will sacrifice system-wide costs. That is, if a vehicle can arrive at some minimum time t , it will do so even at the cost of all flow arriving after t . Ukkusuri showed that the vector M_t satisfying the following inequality enforces the user optimal behaviors:

$$M_t - M_{t-1} > (M_{|T|} - M_t) \cdot TD$$

where TD is the total demands. This inequality implies that the objective would value a transfer from arriving at the destination at time t to arriving at time $t-1$ as being greater value than delaying every vehicle within the network from t until the end of the modeling period. The proof is referred to Ukkusuri (2002). It is noted that the values of the vector M_t can make the resulting LP difficult to solve for realistic, larger-size problems.

The UO DTA formulation shares the same constraints as SO DTA, and has different objective function:

$$\min_{x,y} \sum_{\forall t \in T} \sum_{\forall (i,j) \in E_s} M_t \cdot y_{ij}^t \quad (2.9.1b)$$

Subject to (2.9.2)-(2.9.15)

The dual of UO DTA is (2.10.1)-(2.10.9), (2.10.11)-(2.10.19) and the following (2.10.10b):

$$\lambda_i^{t+1} + \mu_{ij}^{1t} + \mu_{ij}^{3t} \leq M_t, \quad \forall (i, j) \in E_s, \forall t \in T \quad : y_{ij}^t \quad (2.10.10b)$$

The interpretation of the dual variables and the proof of the correctness of the user optimality are given in Ukkusuri (2002).

2.4.2.3 Concise Formulations of DTA LP Models

It was assumed in Ziliaskopoulos (2000) that the source and sink cells have only one adjacent cell, and there is no cell connector that connects a diverging cell and a merging cell. This is not restrictive as the cells can always be structured, so that direct connection between diverging and merging cells is avoided. These assumptions are essential for the basic relationships of CTM and the primal and dual linear programming formulations of SO DTA and UO DTA. However, in this dissertation, we relax these assumptions. That is, the source and sink cells can have more than one adjacent cell, and a diverging cell and a merging cell can be directly connected, so that small test networks can be constructed for the experiment. As such, the concise primal and dual LP formulations of SO DTA and UO DTA must be adopted. Also the definitions of cells and cell connectors in Figure 2.1 fall apart, and we redefine cells and cell connectors as shown in Figure 2.2:

- Source cells (C_R) are cells with at least one successor cell and no predecessor cell ($|FS(i)| \geq 1, |BS(i)| = 0$)
- Sink cells (C_S) are cells with no successor cell and at least one predecessor cell ($|FS(i)| = 0, |BS(i)| \geq 1$)

- Other cells ($C \setminus (C_R \cup C_S)$) are cells with at least one successor cell and at least one predecessor cell ($|FS(i)| \geq 1, |BS(i)| \geq 1$)
- Source cell connectors (E_R) are those with the beginning cell being a source cell
- Sink cell connectors (E_S) are those with the end cell being a sink cell
- Other cell connectors ($E \setminus (E_R \cup E_S)$) are cell connectors that are neither source nor sink cell connectors.

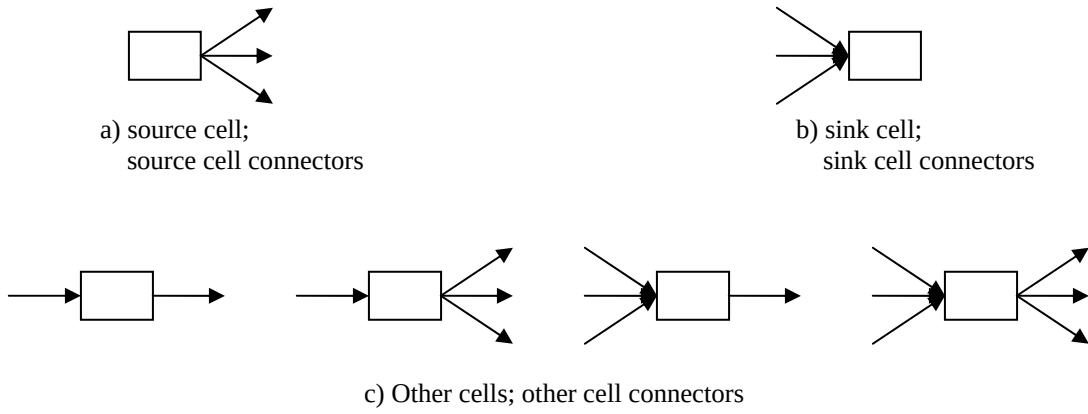


Figure 2.2. Modified Definitions of Cells and Cell Connectors

As such, the concise primal and dual LP formulations of SO DTA are the models (2.11) and (2.12), respectively:

$$\min_{x,y} \sum_{\forall t \in T} \sum_{\forall (i,j) \in E_s} t \cdot y_{ij}^t \quad (2.11.1a)$$

subject to

$$x_i^t - x_i^{t-1} + \sum_{(i,j) \in FS(i)} y_{ij}^{t-1} - \sum_{(j,i) \in RS(i)} y_{ji}^{t-1} = d_i^t \quad \forall i \in C \setminus C_s, t \in T \quad : \pi_i^{0,t} \quad (2.11.2)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^t - x_i^t \leq 0 \quad \forall i \in C \setminus C_s, t \in T \quad : \pi_i^{1,t} \quad (2.11.3)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^t \leq \delta_i^t (N_i^t - x_i^t) \quad \forall i \in C \setminus (C_R \cup C_s), t \in T \quad : \pi_i^{2,t} \quad (2.11.4)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^t \leq Q_i^t \quad \forall i \in C \setminus (C_R \cup C_s), t \in T \quad : \pi_i^{3,t} \quad (2.11.5)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^t \leq Q_i^t \quad \forall i \in C \setminus C_s, t \in T \quad : \pi_i^{4,t} \quad (2.11.6)$$

$$x_i^0 = \zeta_i \quad \forall i \in C \setminus C_s \quad (2.11.7)$$

$$y_{ij}^0 = 0 \quad \forall (i,j) \in E \quad (2.11.8)$$

$$x_i^{[T]} = 0 \quad \forall i \in C \setminus C_s \quad (2.11.9)$$

$$x_i^t \geq 0 \quad \forall i \in C \setminus C_s, t \in T \quad (2.11.10)$$

$$y_{ij}^t \geq 0 \quad \forall (i,j) \in E, t \in T \quad (2.11.11)$$

$$\max_{\pi} \sum_{t \in T} \left(\sum_{i \in C_R} d_i^t \pi_i^{0,t} - \sum_{i \in C \setminus (C_R \cup C_s)} \delta_i^t N_i^t \pi_i^{2,t} - \sum_{i \in C \setminus (C_R \cup C_s)} Q_i^t \pi_i^{3,t} - \sum_{i \in C \setminus C_s} Q_i^t \pi_i^{4,t} \right) \quad (2.12.1)$$

subject to

$$\pi_i^{0,t} - \pi_i^{0,t+1} + \pi_i^{1,t} \leq 0 \quad \forall i \in C_R, t \in T \setminus \{|T|\} \quad : x_i^t \quad (2.12.2)$$

$$\pi_i^{0,t} - \pi_i^{0,t+1} + \pi_i^{1,t} - \delta_i^t \pi_i^{2,t} \leq 0 \quad \forall i \in C \setminus (C_R \cup C_s), t \in T \setminus \{|T|\} \quad : x_i^t \quad (2.12.3)$$

$$\pi_i^{0,t+1} - \pi_j^{0,t+1} - \pi_i^{1,t} - \pi_j^{2,t} - \pi_j^{3,t} - \pi_i^{4,t} \leq 0 \quad \forall (i,j) \in E \setminus E_s, t \in T \setminus \{|T|\} \quad : y_{ij}^t \quad (2.12.4)$$

$$\pi_i^{0,t+1} - \pi_i^{1,t} - \pi_i^{4,t} \leq t \quad \forall (i,j) \in E_s, t \in T \setminus \{|T|\} \quad : y_{ij}^t \quad (2.12.5a)$$

$$\pi_i^{0,t} \text{ u.r.s. } \forall i \in C \setminus C_s, t \in T \quad (2.12.6)$$

$$\pi_i^{1,t}, \pi_i^{4,t} \geq 0 \quad \forall i \in C \setminus C_s, t \in T \quad (2.12.7)$$

$$\pi_i^{2,t}, \pi_i^{3,t} \geq 0 \quad \forall i \in C \setminus (C_R \cup C_s), t \in T \quad (2.12.8)$$

The concise primal LP formulation of UO DTA shares the same constraints as SO DTA with the objective function (2.11.1b):

$$\min_{x,y} \sum_{t \in T} \sum_{(i,j) \in E_s} M_t \cdot y_{ij}^t \quad (2.11.1b)$$

Subject to (2.11.2)-(2.11.11)

The concise dual LP formulation of UO DTA is (2.12.1)-(2.12.4), (2.12.6)-(2.12.8) and the following constraint (2.12.5b).

$$\pi_i^{0,t+1} - \pi_i^{1,t} - \pi_i^{4,t} \leq M_t \quad \forall (i, j) \in E_s, t \in T \setminus \{|T|\} \quad : y_{ij}^t \quad (2.12.5b)$$

The next section describes the simulation-based UO DTA module in VISTA developed by Ziliaskopoulos and Waller (2000) and Chang (2004).

2.4.3 Simulation-based User Optimal Dynamic Traffic Assignment

The functional evaluation in the metaheuristics developed in this dissertation is the UO DTA module in the Visual Interactive System for Transport Algorithms (VISTA). VISTA is an innovative network-enabled framework that integrates spatio-temporal data and models for a wide range of transport applications (Ziliaskopoulos and Waller, 2000). In this study, we employ its DTA module for evaluating different upper-level design decisions. The DTA module in VISTA is a departure-time-based version of the simulation-based UO DTA approach using RouteSim to propagate traffic and satisfy capacity constraints. RouteSim (Ziliaskopoulos and Lee, 1996) is a mesoscopic simulator based on an extension of Daganzo's CTM. The DTA module iteratively employs the time-dependent shortest path algorithm (Ziliaskopoulos and Mahmassani, 1994) to generate vehicle paths, and the inner approximation algorithm for equilibration (Chang, 2004). The implicit assumption is that drivers have perfect information and can divert to alternate paths if it reduces travel time. The inner approximation algorithm can be briefly described as follows.

Inner Approximation Dynamic User Equilibrium Algorithm

Chang (2004) introduced a variational inequality formulation of the single-mode automobile dynamic user equilibrium problem and proposed a solution algorithm, named the Inner Approximation Dynamic User Equilibrium (IADUE) algorithm. The algorithm estimates the equilibrium path assignment using inner approximation methods. Conventionally, vehicle assignment has been performed using the method of successive averages (MSA), which assumes that all previous solutions contribute equally to the final equilibrium, so it assigns vehicles equally among the set of past solutions. In contrast, the IADUE algorithm searches the feasible set of path assignments for the assignment

that minimizes an equilibrium gap function. It searches within a subspace defined by a set or subset of the extreme points of the feasible space. The IADUE algorithm is similar to simplicial decomposition (e.g. Von Hohenbalken, 1977; Hearn et al., 1984), but differs in the descent direction used for each iteration. For simplicial decomposition, the descent direction is the extreme direction that yields the greatest improvement in the gap functional value. For inner approximation, the descent direction is an average of the extreme directions that violate the equilibrium condition.

The notation used in the IADUE algorithm is first described as follows. $G(V, A, T)$ is a network composed of the set of nodes (V), the set of arcs (A) and an assignment period $[0, T]$. d_{rs}^t is the number of automobile trips departing from node r to node s ($r, s \in V$) at time $t \in [0, T]$. P is the set of all spatiotemporal paths from all origins to all destinations. $P(r, s, t) = \{p^1, p^2, \dots, p^\pi\}$ is the set of paths departing at time t from nodes r to s . ξ^{p^k} is the number of automobiles choosing to follow path p^k . Ξ is the vector of path flows; i.e., $\Xi = (\xi^{p^k})$. $\psi^{p^k}(\Xi)$ is the travel time on path p^k . $\Psi(\Xi)$ is the vector of path travel times; i.e., $\Psi(\Xi) = (\psi^{p^k}(\Xi))$.

Since the demand relationships $\sum_{p^k \in P(r, s, t)} \xi^{p^k} = d_{rs}^t$ form a closed, bounded and convex space $D \subset R^\pi$, any assignment Ξ in D is feasible, given that the traffic flow propagation law adopted prevents gridlock and allows all vehicles to complete their trips within time T . In other words, any assignment in D may be selected, provided a sufficient assignment period T such that all assigned vehicles exit the network. A Wardrop equilibrium occurs when no driver can identify a less costly alternative route. That is, at the equilibrium path assignment, any path that is more costly than the minimum path cost remains unused. It is assumed that a driver's selection of an alternative path is a unilateral decision based on the current traffic conditions. Chang (2004) showed that $\psi^{p^x}(\Xi^*) > \psi^{p^y}(\Xi^*)$ implies that $\xi^{p^x} = 0 \quad \forall p^x, p^y \in P(r, s, t) \quad \forall r, s, t$ if and only if $\Psi(\Xi^*)^T \Xi \geq \Psi(\Xi^*)^T \Xi^* \quad \forall \Xi \in D$. Thus, an equilibrium solution, Ξ^* , exists where

$$\Psi(\Xi^*)^T (\Xi - \Xi^*) \geq 0 \quad \forall \Xi \in D \quad (2.13)$$

The inequality (2.13) is a variational inequality (VI) formulation of the dynamic user equilibrium traffic assignment problem. The space D is compact and if $\Psi(\Xi)$ is assumed continuous, then there exists at least one solution to the VI formulation (Chang, 2004). If the cost function $\Psi(\Xi)$ is assumed monotonic such that it satisfies the relationship:

$$(\Psi(\Xi_1) - \Psi(\Xi_2)) \cdot (\Xi_1 - \Xi_2) \geq 0 \quad \forall \Xi_1, \Xi_2 \in D$$

then, the solution is unique.

Chang (2004) argued that traffic network realities, such as signals, prevent the travel time cost function from being truly continuous, and as a result, the existence of an equilibrium point cannot be guaranteed. Moreover, monotonicity may not hold due to temporal interactions of vehicles assigned earlier with those assigned later; thus, the lack of monotonicity precludes the proofs of solution uniqueness and algorithm convergence. Therefore, the proposed algorithm is recognized to operate as a heuristic. When the travel time cost function is assumed to be continuous and monotonic, Chang (2004) proposed an inner approximation to solve the DUE problem. Since the equilibrium conditions defined in the relationship (2.13) are difficult to solve directly, a gap function is formulated such that the optimal solutions of the gap function coincide with points that satisfy the equilibrium conditions. Let Ω be the set of equilibrium solutions that satisfy the VI conditions, then V is a gap function for these conditions if $V(\Xi) = 0 \quad \forall \Xi \in \Omega$ and $V(\Xi) \geq 0 \quad \forall \Xi \in D$. After translating the equilibrium conditions into a gap function, numerical search approaches can be applied. The proposed IADUE algorithm iteratively selects a descent direction in the assignment space D and a step length that minimizes the gap function. The algorithm terminates when the value of the gap function equals to 0 ($V(\Xi) = 0$). The IADUE algorithm is briefly described below.

Step 0: Initialize

Set $n=0$, set link travel time to free flow, and calculate least cost paths for each (r,s,t) .

Set Ξ_0 to all-or-nothing assignment of d_{rs}^t to shortest path for (r,s,t) .

Simulate traffic conditions with assignment Ξ_0 and update link travel times.

Step 1: Choose new solution

Choose descent direction Δ_n .

Select step length $\lambda_n = \arg \min_{\lambda} (V_{Smith}(\Xi_n + \lambda \Delta_n))$ using the golden section search.

Assign demand to $\Xi_{n+1} = \Xi_n + \lambda_n \Delta_n$.

Step 2: Update costs

Simulate traffic conditions with assignment Ξ_n and update link and path travel times.

Step 3: Check for convergence

If $V_{Smith}(\Xi_{n+1}) \geq 0$, set $n = n+1$ and return to Step 1. Otherwise, terminate.

The gap function (V_{Smith}) is based on a gap function originally proposed by Smith (1983) for the static path flow equilibrium problem. Chang (2004) proposed a practical implementation of the IADUE algorithm including three different search phases. Each phase is similar to the above IADUE algorithm, but differs in the gap function and descent direction based on numerical observations. The gap functions and descent directions associated with each phase as well as the detailed implementation can be found in Chang (2004). The next section describes stochastic programming.

2.5 Stochastic Programming

We start by considering the standard form of a two-stage stochastic linear program (SLP2) (Morton, 2004):

$$z_{rp} = z^* = \min_{x \in X} cx + Eh(x, \tilde{\xi}) \quad (2.14)$$

Let the optimal solution of (2.14) be $x_{rp} = x^* \in \arg \min_{x \in X} cx + Eh(x, \tilde{\xi})$. Next, the notations used in this section are described. $X = \{x : Ax = b, x \geq 0\}$ is the deterministic

feasible set. x is the vector of the first-stage decision variables. The recourse function $h(x, \tilde{\xi})$ is a math program:

$$h(x, \tilde{\xi}) = \min_{y \geq 0} \tilde{f}y \quad (2.15.1)$$

$$\text{subject to } \tilde{D}y = \tilde{B}x + \tilde{d} . \quad (2.15.2)$$

$\tilde{\xi}$ is a vector of discrete random variables $\tilde{\xi} = \text{vec}(\tilde{D}, \tilde{d}, \tilde{B}, \tilde{f})$ where the “vec” operator reads its arguments column-wise to form a single vector. y is the vector of the second-stage decision variables. Ω is the set of all possible realizations of $\tilde{\xi}$.

If $\tilde{D} = D$ is deterministic, the SLP2 has fixed recourse. In the SLP2 of the single-level SO and UO NDP LP models in Chapter 3, if the saturation flows, jam densities and time-varying demands are stochastic, these correspond to the parameters in the right-hand side of the constraints are random; i.e. in the standard form $\tilde{\xi} = \text{vec}(\tilde{d}, \tilde{B})$. Then, $h(x, \tilde{\xi})$ is convex on the support of $\tilde{\xi}$ ($\Xi_{\tilde{\xi}}$), and Jensen’s inequality results in $Eh(x, \tilde{\xi}) \geq h(x, E\tilde{\xi})$ (see Proposition 3.1 in Kall and Wallace ,1994), which can be shown as follows. First, let’s define π as the dual variables associated with the problem (2.15) and considering the dual program of the recourse function (2.15):

$$\begin{aligned} h(x, E\tilde{\xi}) &= \max_{\pi D \leq f} \pi((E\tilde{B})x + E\tilde{d}) = \max_{\pi D \leq f} \sum_{\omega \in \Omega} p^\omega \pi(\tilde{B}^\omega x + \tilde{d}^\omega) = \max_{\substack{\pi^\omega D \leq f \forall \omega \in \Omega \\ \pi^\omega = \pi \forall \omega \in \Omega}} \sum_{\omega \in \Omega} p^\omega \pi^\omega (\tilde{B}^\omega x + \tilde{d}^\omega) \\ &\leq \max_{\pi^\omega D \leq f \forall \omega \in \Omega} \sum_{\omega \in \Omega} p^\omega \pi^\omega (\tilde{B}^\omega x + \tilde{d}^\omega) = \sum_{\omega \in \Omega} \left(\max_{\pi^\omega D \leq f} p^\omega \pi^\omega (\tilde{B}^\omega x + \tilde{d}^\omega) \right) \\ &= \sum_{\omega \in \Omega} \left(p^\omega \max_{\pi^\omega D \leq f} \pi^\omega (\tilde{B}^\omega x + \tilde{d}^\omega) \right) = Eh(x, \tilde{\xi}) . \end{aligned} \quad (2.16)$$

Consequently, the deterministic NDP models employing the mean values of parameters yield underestimated solutions to the stochastic models. Note that if $\tilde{\xi} = \text{vec}(\tilde{D}, \tilde{d}, \tilde{B}, \tilde{f})$, $h(x, \tilde{\xi})$ is not convex on $\Xi_{\tilde{\xi}}$, and Jensen’s inequality cannot be applied.

Morton (2004) discussed three possible solution methodologies for an SLP2 model. First, when the size of the problem in terms of the number of scenarios $|\Omega|$ is very small, we may solve the large-scale SLP2 model for an exact solution by direct

methods such as simplex and the L-shaped method. If $|\Omega|$ is large, or $\tilde{\xi}$ is a continuous vector, then performing an exact function evaluation of $H(x) = Eh(x, \tilde{\xi})$ may be difficult or impossible unless the subproblem has some very special structure like simple recourse. This leads to the second methodology, Sequential Approximation Method (SAM), which estimates a solution with a deterministic optimality gap. SAM employs Jensen's inequality and the Edmundson-Madansky (E-M) inequality to respectively determine lower and upper bound with the assumption that $\tilde{\xi} = \text{vec}(\tilde{d}, \tilde{B})$. The Jensen bound requires just one subproblem be solved, regardless of the dimension of $\tilde{\xi} = (\tilde{\xi}_1, \dots, \tilde{\xi}_l)$; however, the E-M bound requires 2^l subproblem solutions. Thus, the computational effort grows exponentially in the dimension of $\tilde{\xi}$. Kall and Wallace (1994) indicated that SAM can efficiently solve an SLP2 model when the number of stochastic variables are at most 10; i.e. $l \leq 10$. Further, when $\tilde{\xi} = \text{vec}(\tilde{D}, \tilde{d}, \tilde{B}, \tilde{f})$, Jensen's and the E-M inequality do not apply, and neither do SAM. To overcome these limitations, Mak et al. (1999) suggested the Monte Carlo bounding techniques to solve the SLP2 model. The Monte Carlo bounding techniques can solve difficult and very large problems, but yields probabilistic optimality gaps. Karoonsoontawong and Waller (2005) employed the Monte Carlo bounding techniques to compare the two SLP2 NDP models based on SO and UO DTA models when accounting for both demand and supply uncertainties. In this dissertation, all proposed SLP2 models in Chapter 3 are solved by direct methods as the illustrative examples are sufficiently small in order to help us gain insight into the problem.

It is noted that there is the chance constrained programming (CCP), another scheme of stochastic programming. The pros and cons of SLP2 and CCP are discussed by Morton (2004) as follows. The advantages of the chance-constrained model are that the model is not driven by the worst scenario, and some applications are naturally posed this way. However, the magnitude of violation is not captured, and the feasible region is not always convex. The advantage of the SLP2 model with recourse is that it captures decision-maker's ability to take corrective action. However, the objective function

employs only the first moment. One can capture the higher moments in the objective function by employing the robust optimization scheme discussed in the next section.

2.6 Robust Optimization

One criticism of SLP2 models is that it only optimizes only the first moment of the distribution of objective value. Thus, the optimal solution may yield the least expected cost but with wildly different costs across various scenarios of uncertain input parameters. This inspired Mulvey et al. (1995) to propose their robust optimization model. It incorporated the higher moments into the objective function of SLP2, and the objective function becomes a weighted summation of the first moment and the higher moments of the distribution of objective value. That is, the robust optimization model integrates the methods of multiobjective programming into stochastic optimization. The importance of controlling variability of the solution as opposed to just optimizing its first moment is well recognized in other applications such as portfolio management (Markowitz, 1959), outsourcing in manufacturing (Escudero et al., 1993), capacity planning model (Paraskevopoulos et al., 1991) and uncapacitated network design problems (Gutierrez et al. 1996).

The canonical form of a robust optimization (RO) model is extended from that of SLP2:

$$z_{RO} = \min_{x \in X} cx + Eh(x, \tilde{\xi}) + \lambda \cdot \rho(h(x, \tilde{\xi})) \quad (2.17)$$

where x , X , $h(x, \tilde{\xi})$ are defined in the same way as those in SLP2. λ is a goal programming weight used to find a range of answers that tradeoff the higher moments for the first moment. $\rho(\cdot)$ is the higher moment function that is a measure of risk. If $\rho(\cdot)$ or λ is set to zero, the robust optimization model essentially becomes the SLP2 model. The measure of risk can be set differently depending on applications. One possible functional form is the maximum value as used in the worst-case analysis:

$$\rho(h(x, \tilde{\xi})) = \max_{\omega \in \Omega} h(x, \tilde{\xi}^\omega) \quad (2.18.1)$$

Another is the variance:

$$\rho(h(x, \tilde{\xi})) = V[h(x, \tilde{\xi})] \quad (2.18.2)$$

Then, the objective function is a utility function that represents a tradeoff between mean value and variability. The appropriate functional form of $\rho(\cdot)$ is problem dependent, and it also implies the proper solution algorithm. For example, when $\rho(h(x, \tilde{\xi})) = V[h(x, \tilde{\xi})]$, the resulting model is a quadratic program.

While the robust optimization has some advantages over the stochastic optimization, it does have limitations. First, a robust optimization model is a parametric program, and there is no a priori procedure for specifying a correct choice of the goal programming weight λ . This problem is prevalent in multi-objective programming (Keeney and Raiffa, 1976). Second, the robust optimization as well as the stochastic optimization does not provide a mechanism for specifying the best set of realizations of the uncertain parameters. Mulvey et al. (1995) suggested that the methods for integrating the variance reduction techniques into stochastic optimization (e.g. Infanger, 1992) also apply to robust optimization.

It is noted that there are various robust optimization concepts in the literature, and we adopt the concept by Mulvey et al. (1995) as it can capture the tradeoff between the expected cost and the expected risk in NDP, SSD and NDP-SSD. The next section describes the linear bi-level programming problem.

2.7 Linear Bi-level Optimization

In this dissertation, we propose the new linear bi-level program for NDP. This section provides essential basics on linear bi-level programming, which greatly differs from the well-known single-level linear programming. Readers should understand this concept in order to follow the new bi-level formulation and exact solution methods in Chapter 3.

Bard (1998) indicated that there had been nearly two dozen algorithms proposed for solving the linear bi-level programming problem (L-BLPP), but only some of them appear more successful. He classified the viable algorithms into three categories: vertex enumeration, Karush-Kuhn-Tucker (KKT) condition-based approach and penalty

approach. This study employs two algorithms that belong to the first two categories. As an essential basis to understand the two algorithms, the underlying theoretical properties of the L-BLPP are first discussed in the following standard form (Bard, 1998) for convenient discussion:

$$\min_{x \in X} F(x, y) = c_1 x + d_1 y \quad (2.19.1)$$

$$\text{subject to} \quad A_1 x + B_1 y \leq b_1 \quad (2.19.2)$$

$$\min_{y \in Y} f(x, y) = c_2 x + d_2 y \quad (2.19.3)$$

$$\text{subject to} \quad A_2 x + B_2 y \leq b_2 \quad (2.19.4)$$

where $x \in X \subset R^n$, $y \in Y \subset R^m$, $F : X \times Y \rightarrow R^1$, $f : X \times Y \rightarrow R^1$, $c_1, c_2 \in R^n$, $d_1, d_2 \in R^m$, $b_1 \in R^p$, $b_2 \in R^q$, $A_1 \in R^{p \times n}$, $B_1 \in R^{p \times m}$, $A_2 \in R^{q \times n}$, $B_2 \in R^{q \times m}$. The sets X and Y place additional restrictions on the variables; specifically, $X = \{x : x \geq 0\}$ and $Y = \{y : y \geq 0\}$.

Next, five fundamental definitions (Bard, 1998) are given:

(a) Constraint region of the L-BLPP:

$$S = \{(x, y) : x \in X, y \in Y, A_1 x + B_1 y \leq b_1, A_2 x + B_2 y \leq b_2\}$$

(b) Feasible set for the follower for each fixed $x \in X$:

$$S(x) = \{y \in Y : B_2 y \leq b_2 - A_2 x\}$$

(c) Projection of S onto the leader's decision space:

$$S(X) = \{x \in X : \exists y \in Y, A_1 x + B_1 y \leq b_1, A_2 x + B_2 y \leq b_2\}$$

(d) Follower's rational reaction set for $x \in S(X)$:

$$P(x) = \{y \in Y : y \in \arg \min [f(x, \hat{y}) : \hat{y} \in S(x)]\}$$

(e) Inducible region:

$$IR = \{(x, y) : (x, y) \in S, y \in P(x)\}$$

The rational reaction set $P(x)$ defines the response, whereas the inducible region IR represents the set over which the leader may optimize. Thus, the L-BLPP (2.19) can be represented as the implicit representation: $\min\{F(x, y) : (x, y) \in IR\}$.

Subsequently, we state five theoretical properties of L-BLPP (see proofs in Bard, 1998):

Fact 0: The L-BLPP is strongly NP-hard.

Fact 1: IR can be written equivalently as a piecewise linear equality constraint comprised of supporting hyperplanes of S .

Fact 2: The L-BLPP (2.19) is equivalent to minimizing F over a feasible region comprised of a piecewise linear equality constraint.

Fact 3: A solution to the L-BLPP occurs at a vertex of IR .

Fact 4: The solution (x^*, y^*) of the L-BLPP occurs at a vertex of S .

These theoretical properties of L-BLPP are employed in the exact analytical approach for the DTA-based network design model in Chapter 3. The next section describes a special case of mixed integer L-BLPP when the upper-level decision variables are binary, and the lower-level variables are continuous.

2.8 Zero-One Continuous Linear Bi-level Programming

In this dissertation, we propose the new mixed-zero-one continuous bi-level programs for SSD and NDP-SSD. This section provides essential basics for developing the exact solution methods of these models. Specifically, we discuss the reduction of the zero-one continuous linear bi-level programming problem (ZC-L-BLPP) to an equivalent parametric linear bi-level program.

The ZC-L-BLPP can be stated below (Bard, 1998).

$$\min_{x \in X} F(x, y) = c_1 x + d_1 y \quad (2.20.1)$$

$$\text{subject to} \quad A_1 x + B_1 y \leq b_1 \quad (2.20.2)$$

$$\min_{y \in Y} f(y) = d_2 y \quad (2.20.3)$$

$$\text{subject to} \quad A_2 x + B_2 y \leq b_2 \quad (2.20.4)$$

where $c_1 \in R^n$, $d_1, d_2 \in R^m$, $b_1 \in R^p$, $b_2 \in R^q$, $A_1 \in R^{p \times n}$, $B_1 \in R^{p \times m}$, $A_2 \in R^{q \times n}$, $B_2 \in R^{q \times m}$, $X = \{0,1\}^n$, $Y \subseteq R^m$.

Bard (1998) showed that the ZC-L-BLPP (2.20) is equivalent to an L-BLPP, using the following Fact 5:

Fact 5: There exists a positive real M such that the ZC-L-BLPP (2.20) and the following parametric L-BLPP, L-BLPP(M) (2.21), have the same optimal solutions:

$$\min_x F(x, y) = c_1x + d_1y + Me_x^T u \quad (2.21.1)$$

$$\text{subject to} \quad A_1x + B_1y \leq b_1 \quad (2.21.2)$$

$$0 \leq x \leq e_x \quad (2.21.3)$$

$$\min_{y, u} d_2y - e_x^T u \quad (2.21.4)$$

$$\text{subject to} \quad A_2x + B_2y \leq b_2 \quad (2.21.5)$$

$$u \leq x \quad (2.21.6)$$

$$u \leq e_x - x \quad (2.21.7)$$

where e_x is the vector of ones with dimension n .

The proof of Fact 5 utilizes the below Fact 6, which is a generalization of a result established by Kalantari and Rosen (1982) for integer linear programs.

Fact 6: If θ is a concave function, there exists a positive real number M such that the problems (2.22) and (2.23) have the same solutions:

$$\min_{x \in X} c_1x + d_1y \quad (2.22.1)$$

$$\text{subject to} \quad (x, y) \in P \quad (2.22.2)$$

where $P = \bigcup_{i=1}^l P_i$, P_i is a polyhedral set for $i = 1, \dots, l$ and l is a positive integer.

$$\min c_1x + d_1y + M\theta(x) \quad (2.23.1)$$

$$\text{subject to} \quad 0 \leq x \leq e_x \quad (2.23.2)$$

$$(x, y) \in P \quad (2.23.3)$$

where $\theta: R^n \rightarrow R$ is a continuous function such that $\theta(x) \geq 0$ for all $0 \leq x \leq e_x$ and $\theta(x) = 0$ if and only if $x \in X$.

The detailed proofs of Facts 5 and 6 are referred to Bard (1998) and Kalantari and Rosen (1982). This reduction technique is employed in the exact solution methods for the deterministic, stochastic and robust bi-level SSD and NDP-SSD models in Chapter 3. The next section reviews four metaheuristic methods adopted in this study.

2.9 Metaheuristics

A metaheuristic refers to a master procedure that guides and modifies other heuristics to produce solutions beyond those that are normally generated for a local optimum (Glover and Laguna, 1997). The heuristics guided by such a metaheuristic may be high-level algorithms or a simple description of available moves from the current solution to the next. The metaheuristics outperform the gradient search algorithms on many difficult optimization problems involving discontinuous, noisy, high-dimensional and multimodal objective functions (Goldberg, 1989). When a difficult problem (e.g. NP-hard), is solved by metaheuristics, one has to accept a sub-optimal solution within the allotted CPU time. Naturally, specific task-dependent termination criteria can be introduced such as a threshold on the quality of the solution (Battiti and Tecchiolli, 1994). However, in this study, we wish to compare the efficiency and effectiveness of different metaheuristic methods; thus the employed stopping criterion is the maximum number of objective functional evaluations (also called trial). Four metaheuristics considered in this paper are random search (RS), simulated annealing (SA), genetic algorithm (GA) and reactive tabu search (RTS). It is noted that these employ the same functional evaluation, the simulation-based UO DTA module in VISTA.

Random Search

The random search (RS) is the simplest stochastic global optimization method (Brooks, 1958; Boender and Romeijn, 1995). The algorithm randomly generates a sequence of feasible solutions and keeping track of the best solution.

Simulated Annealing

The simulated annealing (SA) algorithm is motivated by an analogy to the statistical mechanics of annealing of solids (Kirkpatrick et al., 1983). The system is said to be in thermal equilibrium at a temperature T if the probability of being in state i with energy E_i is governed by a Boltzman distribution:

$$\text{Prob}(E_i) = \exp(-E_i / k_B T) / \sum_j \exp(-E_j / k_B T)$$

where k_B is the Boltzman constant. The annealing process leads to this probability law for energy states. Metropolis et al. (1953) developed an algorithm to simulate the behavior of a collection of atoms at thermal equilibrium at a particular temperature. By repeating the basic steps of this algorithm many times, the accepted configurations form a Boltzman distribution at a particular temperature. The algorithm of Metropolis et al. (1953) laid a basis for SA for combinatorial optimization. A particular configuration for the annealing of physical systems is analogous to the vector of variables, and the energy is analogous to its objective functional value for minimization problem. SA for continuous variables was originally proposed by Vanderbilt and Louie (1984), and was applied to transportation engineering by Friesz et al. (1992) and Lee (1998), to name a few. The SA in Chapter 4 is developed based on the work by Lee (1998) and Vanderbilt and Louie (1984).

Genetic Algorithm

Inspired by Darwin's theory of survival of fittest during evolution, the genetic algorithm (GA) is an iterative procedure that maintains a population of candidate solutions to the objective function (Goldberg, 1989; Michalewicz, 1996; Grefenstette, 1990). The basic concepts of GA were developed by Holland (1975) and his students. Holland (1975) and DeJong (1975) showed that GA provides a highly efficient heuristic for information gathering in complex search spaces and exhibits impressive efficiency in

practice. GA in transportation applications can be found, for example, in Ceylan and Bell (2004) and Teklu (2003).

GA can generally be described as follows (Grefenstette, 1990). Let $Pop(k)$ be the binary-string population from generation k . There are N_{POP} chromosomes per a population, and let $fr^{(j;k)}$ represent chromosome j of length m in the population from generation k for all $j = 1, 2, \dots, N_{POP}$. Thus, the population from iteration k can be written as: $Pop(k) = \{fr^{(1;k)}, \dots, fr^{(N_{POP};k)}\}$. Each chromosome $fr^{(j;k)}$ represents a vector of decision variables, but the meaning associated with each chromosome is unknown to the GA. During each generation, the current population is evaluated, and a new population of candidate solutions is formed on the basis of this evaluation. Note that the terms used interchangeably in this dissertation are chromosome, string, individual, structure and solution. The GA in Chapter 4 is developed using the source code by Grefenstette (1990).

Reactive Tabu Search

The tabu search has been shown to be an effective and efficient procedure for combinatorial optimization (Battiti and Tecchiolli, 1994). It integrates a hill-climbing search technique, which is based on a set of elementary moves, and a heuristic to avoid the stops at local optima and the occurrence of cycles. The tabu search was initially created with a constant tabu tenure by Glover (1989). The proper choice of tabu tenure is critical to the success of the algorithm. The tabu tenure should be sufficiently long to prevent cycles but short enough such that the search is not overly constrained (Battiti and Tecchiolli, 1994). The overview of tabu search can be found in Glover (1989, 1990).

As a variation of tabu search, the reactive tabu search (RTS) was proposed by Battiti and Tecchiolli (1994). They made an analogy between the evolution of the direct search process and the theory of dynamical systems (Serra and Zanarini, 1990; Shuster, 1984). The direct search procedure keeps updating the current solution in order to find the global optimum. The search trajectory in the solution space is subject to the movements dictated by the search scheme. Battiti and Tecchiolli argued that three

possibilities could prevent the direct search from finding the global optimum. First, there are possibly multiple local minima, called attractors of the system dynamics in the theory of dynamical systems. These local minima are fixed points in the solution space. As a kind of direct search techniques, the steepest descent algorithm always terminates at a local minimum. To avoid trapping at a local optimum, one must introduce a search scheme such as RTS that forces the search to exit from the local minimum and continue the exploration. Second, limit cycles or closed orbits can cause the search trajectory to endlessly repeat a sequence of solutions. RTS discourages the occurrence of such cycles by dynamically updating the tabu tenure according to the status of the search. The third possibility is when fixed points and limit cycles are not present, but the search trajectory is confined in a limited region of the solution space. If this limited region does not contain the global minimum, the search will never find it. This phenomenon, called deterministic chaos, can be described by the concept of chaotic attractor in the theory of dynamical systems. Chaotic attractors are characterized by a contraction of the areas, so that trajectories starting with different initial solutions will be confined in a limited area of the configuration space, and by a sensitive dependence upon the initial solutions, so that different trajectories will diverge. A detailed theoretical analysis of chaotic attractor is referred to Shuster (1984). That is, the search trajectory appears random, but in fact, it is deterministic. The avoidance of cycles and confinements assures that the available time is spent in an efficient exploration of the search space, and specific termination conditions apart from the expiration of the given CPU time are not necessary. In this dissertation, we modify RTS to solve NDP, SSD and NDP-SSD.

2.10 Summary

The works reviewed in this chapter form the foundation for the research in this dissertation, which focuses on development of analytical and simulation-based approaches to solving the robust optimization model of the combined network design, signal optimization and dynamic traffic assignment when accounting for uncertainty parameters. First, three transportation network improvement optimization problems are

reviewed: 1) network design problem, 2) traffic assignment based signal optimization, and 3) combined network design, signal optimization and traffic assignment. Next, relevant cell transmission-based dynamic traffic assignment schemes, including linear programming models and simulation-based procedure, are discussed. Then, two mathematical programming techniques dealing with uncertain input data are described: stochastic programming and robust optimization. Further, the linear bi-level optimization is discussed. Also, as a special case of mixed-integer linear bi-level optimization, the zero-one continuous linear bi-level programming is described. Lastly, an overview of relevant metaheuristic methods is presented. The next chapter describes the analytical approach to tackle the three transportation network improvement optimization problems.

Chapter 3. Analytical Approach

This chapter develops a core set of mathematical models that can allow new perspectives on traditionally difficult and intractable problems, namely, continuous network design problem (NDP), DTA-based signal setting design problem (SSD) and the combined NDP, SSD and DTA problem (NDP-SSD). All three problems are bi-level by nature. That is, the transportation planners make different design decisions to optimize the system, whereas the transportation users react to the design decisions such that the dynamic user equilibrium is realized. We further increase the problem complexity by considering the uncertainty of the time-dependent origin-destination (O-D) demands due to the inherent stochastic nature of forecasted traveler demand. Then, the problems are naturally modeled as two-stage stochastic programs. That is, the planners make their design decisions, considered as the first-stage decision variables, before the O-D demands are realized; subsequently, the transportation users make their route choice decisions, considered as the second-stage decision variables. Furthermore, to better deal with the uncertainty, we aim at optimizing not only the expected cost but also the expected risk. Then, the problems are cast into the robust optimization problems. It is noted that with such complexity these models are not developed for final applications and are limited to solving a small network; however, they enable us to gain insights into the problems and facilitate the progressive development towards the practical methodologies.

All models in this chapter are limited to fixed-departure-time single-destination O-D demands due to the underlying DTA LP models discussed in Chapter 2. In NDP, we assume that continuous investment variables change practical capacity but not free flow travel time, and the improvement cost is a linear function of the improvement levels. It is noted that the convex piecewise linear improvement cost functions can easily be incorporated. In SSD, we optimize cycle lengths, green splits, offsets, and phase sequencing with assumed fixed signal phasing. These five signal setting variables constitute a signal timing plan. We also assume that all possible signal timing plans for all intersections are enumerated. The optimization models select the timing plans for all

intersections such that the total system travel time (TSTT) is minimized subject to dynamic traffic assignment conditions. Also, we allow intersections to have different cycle times, so our models are less constrained than those assuming a common cycle length in the literature. It is noticed that our SSD models exploit the benefit of signal coordination since time offsets are optimized. In NDP-SSD, the assumptions of both NDP and SSD are applied.

The definitions of sets, parameters and variables are first provided, followed by the NDP models, the SSD models and the NDP-SSD models. Then, the exact solution methods for the three problems are described.

Notation

Sets

C	= set of cells
C_S	= set of sink cells
C_R	= set of source cells
T	= set of discrete time intervals
E	= set of cell connectors
E_S	= set of sink cell connectors
$FS(i)$	= set of cell connectors emanating from cell i
$RS(i)$	= set of cell connectors emanating to cell i
$INTERSECTION$	= set of intersection cells
$P(i,j)$	= set of phases which allow flow from cell i to intersection cell j
STP_i	= set of signal timing plans for intersection cell i
P_i	= set of all phases for intersection cell i

Parameters

TAB	= total available budget
δ_i^t	= ratio of link free flow speed and backward propagation speed for each

	cell and time interval (= 1 in this paper)
M_t	= cost per time interval that yields user optimal flows
ζ_i	= initial number of vehicles in cell i (= 0 in this paper)
d_i^t	= demand at cell i in time interval t
N_i^t	= maximum number of vehicles in cell i at time interval t
Q_i^t	= maximum number of vehicles that can flow into or out of cell i during time interval t
χ_i	= increase in N_i^t per one unit of b_i (= χ in this paper)
ϕ_i	= increase in Q_i^t per one unit of b_i (= ϕ in this paper)
$phase(i,t)$	= phase of signal timing plan i at time t

Note: A parameter with tilde sign is a stochastic parameter. In this study, we consider only one uncertain parameter, namely, $\tilde{d}_i^t$ (stochastic time dependent demand), which are assumed independent across time intervals and cells.

$$\tilde{\xi} = \text{vec}(\tilde{d}_i^t \forall i \in C, t \in T) = \text{vector of stochastic parameters}$$

Variables

b_i	= amount of budget allocated to cell i
x_i^t	= number of vehicles in cell i at time interval t
y_{ij}^t	= number of vehicles moving from cell i to cell j at time interval t
s_i	= 1 if the signal timing plan i is chosen, and 0 otherwise
p_{ij}^t	= 1 if flow is allowed from cell i to j at time t , and 0 otherwise

3.1 NDP Models

Waller (2000) formulated the deterministic system optimal network design model (SONDP) and the two-stage stochastic SONDP model (SLP2-SONDP) based on the SO DTA LP formulation reviewed in Chapter 2. Then, Ukkusuri (2002) formulated the

deterministic user optimal network design model (UONDP) and the two-stage stochastic UONDP model (SLP2-UONDP) based on the UO DTA LP model in Chapter 2. In this study, we propose the deterministic bi-level programming NDP model (BLPNDP), the corresponding two-stage stochastic program (SLP2-BLPNDP) and the robust bi-level programming NDP model (RO-BLPNDP). For the stochastic and robust models, only the O-D demands are considered uncertain with known probability distribution.

3.1.1 Deterministic NDP Models (SONDP, UONDP and BLPNDP)

The three deterministic NDP models (SONDP, UONDP and BLPNDP) share the same set of constraints. The first set is the budget constraints (3.1.1)-(3.1.2), and the second is the cell transmission model (CTM) constraints (3.1.3)-(3.1.12).

Let $BUDGET_CONSTRAINT(b)$ denote Constraints (3.1.1)-(3.1.2):

$$\sum_{i \in C \setminus C_S} b_i \leq TAB \quad (3.1.1)$$

$$b_i \geq 0 \quad \forall i \in C \setminus C_S \quad (3.1.2)$$

The network improvements are constrained by a user-specified total available budget (TAB). Variable b_i is defined as the amount of the budget spent on cell i to increase its saturation flow by $\phi_i \cdot b_i$ and its jam density by $\chi_i \cdot b_i$. This allows the model to compute improvements for street segments smaller than a link. However, this degree of flexibility is not considered in this study. It will be seen in Chapter 5 that the budgets spent throughout a particular link are always identical due to the underlying traffic flow theoretical CTM model; otherwise, a bottleneck that causes traffic congestion can take place on the link.

Next, denote by $CTM(x, y, b)$ Constraints (3.1.3)-(3.1.12):

$$x_i^t - x_i^{t-1} + \sum_{(i,j) \in FS(i)} y_{ij}^{t-1} - \sum_{(j,i) \in RS(i)} y_{ji}^{t-1} = d_i^t \quad \forall i \in C \setminus C_S, t \in T \quad (3.1.3)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^t - x_i^t \leq 0 \quad \forall i \in C \setminus C_S, t \in T \quad (3.1.4)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^t \leq \delta_i^t (N_i^t + \chi_i \cdot b_i - x_i^t) \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad (3.1.5)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^t \leq Q_i^t + \phi_i \cdot b_i \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad (3.1.6)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^t \leq Q_i^t + \phi_i \cdot b_i \quad \forall i \in C \setminus C_S, t \in T \quad (3.1.7)$$

$$x_i^0 = \zeta_i \quad \forall i \in C \setminus C_S \quad (3.1.8)$$

$$y_{ij}^0 = 0 \quad \forall (i, j) \in E \quad (3.1.9)$$

$$x_i^{|T|} = 0 \quad \forall i \in C \setminus C_S \quad (3.1.10)$$

$$x_i^t \geq 0 \quad \forall i \in C \setminus C_S, t \in T \quad (3.1.11)$$

$$y_{ij}^t \geq 0 \quad \forall (i, j) \in E, t \in T \quad (3.1.12)$$

Parameter χ_i specifies the increase in the jam density of cell i for one unit of TAB , while ϕ_i denotes the increase in the saturation flow of cell i for one unit of TAB . In the DTA model, N_i^t that appears in Constraint (2.11.4) is replaced with $N_i^t + \chi_i \cdot b_i$, and Q_i^t that appears in constraints (2.11.5) and (2.11.6) is replaced with $Q_i^t + \phi_i \cdot b_i$. Two basic traffic flow relationships are embedded in the CTM constraints. The first is the cell mass conservation (3.1.3). The second states that the traffic flow between two cells is constrained by the number of vehicles occupying the upstream cell (3.1.4), the remaining capacity of the downstream cell (3.1.5), and the maximum flow that can get out of the upstream cell and into the downstream cell (3.1.6)-(3.1.7). One important constraint (3.1.10) enforces that all vehicles must reach the destination by the last time interval. Without this constraint, the optimal solution is erroneous; i.e. all vehicles do not leave the origin. For each cell, the constants N_i^t , Q_i^t and δ_i^t can vary with time to capture time-dependent capacity and flow variations. Daganzo (1994) showed that these constants together with the length of the cell can reasonably capture the road characteristics. Typically, these constants should be obtained from the calibration using loop detector data (Munoz et al., 2004). However, for simplicity, we arbitrarily assume the values of N_i^t and Q_i^t for our test problems, and set all δ_i^t values equal to 1.

SONDP Model

The objective of the SONDP is to determine the optimal network improvements so as to minimize the TSTT under SO DTA behavior. The linear continuous SONDP model is (Waller et al., 2006):

$$\min_{b,x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^t) \quad (3.2.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.2.2)$$

$$CTM(x, y, b) \quad (3.2.3)$$

It is noted that TSTT basically equals the difference between the summation of arrival times (*ARRIVAL*) at the sink cell and the summation of departure times (*DEPARTURE*) from the source cells. *ARRIVAL* is $\sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^t)$. Since the underlying DTA LP model

assumes fixed-departure-time O-D demands, *DEPARTURE* is constant. Thus, *DEPARTURE* can be dropped from the optimization model without an effect on the optimal solution. In our analytical studies (Chapters 3 and 5), we refer to TSTT as $\sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^t)$.

UONDP Model

The objective of the UONDP is the same as that of UO DTA LP model. Thus, it is to determine the optimal network improvements such that the individual users minimize their own travel time. It is noted that the TSTT is not optimized in this model. The linear continuous UONDP model is (Ukkusuri, 2002):

$$\min_{b,x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} (M_t \cdot y_{ij}^t) \quad (3.3.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.3.2)$$

$$CTM(x, y, b) \quad (3.3.3)$$

A common drawback of both SONDP and UONDP models is that they cannot represent the problem correctly. As discussed previously, the nature of the NDP is bi-level. Thus, the improved deterministic model is the BLPNDP described next.

BLPNDP Model

The objective of the BLPNDP is to determine the optimal network improvements such that the TSTT is minimized under UO DTA behavior. The linear continuous bi-level model is (Karoonsontawong and Waller, 2006):

$$\min_b \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^t) \quad (3.4.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.4.2)$$

$$\min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} (M_t \cdot y_{ij}^t) \quad (3.4.3)$$

$$\text{subject to} \quad CTM(x, y, b) \quad (3.4.4)$$

The three deterministic NDP models assume all parameters are known and deterministic. These are obviously unrealistic as the parameters, especially O-D demands, are uncertain. The improved approach to model the NDP is the two-stage stochastic linear programming (SLP2) described in the next section.

3.1.2 Two-Stage Stochastic NDP Models (SLP2-SONDP, SLP2-UONDP and SLP2-BLPNDP)

The three two-stage stochastic programs (SLP2-SONDP, SLP2-UONDP and SLP2-BLPNDP) and the robust model (RO-BLPNDP) employ the budget constraints (3.1.1)-(3.1.2) and the following CTM constraints corresponding to each realization ω of uncertain parameters (i.e. O-D demands).

Let $CTM^\omega(x, y, b)$, $\forall \omega \in \Omega$ denote Constraints (3.5.1)-(3.5.10):

$$x_i^{t,\omega} - x_i^{t-1,\omega} + \sum_{(i,j) \in FS(i)} y_{ij}^{t-1,\omega} - \sum_{(j,i) \in RS(i)} y_{ji}^{t-1,\omega} = \tilde{d}_i^{t,\omega} \quad \forall i \in C \setminus C_S, t \in T \quad (3.5.1)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^{t,\omega} - x_i^{t,\omega} \leq 0 \quad \forall i \in C \setminus C_S, t \in T \quad (3.5.2)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^{t,\omega} \leq \delta_i^t (N_i^t + \chi_i \cdot b_i - x_i^{t,\omega}) \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad (3.5.3)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^{t,\omega} \leq Q_i^t + \phi_i \cdot b_i \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad (3.5.4)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^{t,\omega} \leq Q_i^t + \phi_i \cdot b_i \quad \forall i \in C \setminus C_S, t \in T \quad (3.5.5)$$

$$x_i^{0,\omega} = \zeta_i \quad \forall i \in C \setminus C_S \quad (3.5.6)$$

$$y_{ij}^{0,\omega} = 0 \quad \forall (i,j) \in E \quad (3.5.7)$$

$$x_i^{|T|,\omega} = 0 \quad \forall i \in C \setminus C_S \quad (3.5.8)$$

$$x_i^{t,\omega} \geq 0 \quad \forall i \in C \setminus C_S, t \in T \quad (3.5.9)$$

$$y_{ij}^{t,\omega} \geq 0 \quad \forall (i,j) \in E, t \in T \quad (3.5.10)$$

Next, the standard two-stage linear stochastic network design model is:

$$\min_b E[h(b, \tilde{\xi})] = \sum_{\omega \in \Omega} [p^\omega \cdot h(b, \tilde{\xi}^\omega)] \quad (3.6.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.6.2)$$

According to Morton (2004), the timing of what we know and when we know it is of fundamental importance for SLP2 and RO. Here, we articulate the order of the events for clarity. The first-stage decisions are the amount of budget allocated to each cell i (b_i), and are made with the knowledge of the distribution of stochastic parameters. Thereafter, a realization of stochastic parameters ($\tilde{d}_i^{t,\omega}$ of $\tilde{d}_i^t$) unfolds. After that, the scenario-dependent second-stage decisions are determined from corresponding DTA; i.e., the number of vehicles assigned to cell i at time t for scenario ω ($x_i^{t,\omega}$) and the number of vehicles traversing on cell connectors between cell i and cell j at time t for scenario ω ($y_{ij}^{t,\omega}$). Note that the first-stage decisions cannot depend on the scenario ω because this would correspond to knowing the future. The timing of decisions is illustrated in Figure 3.1.

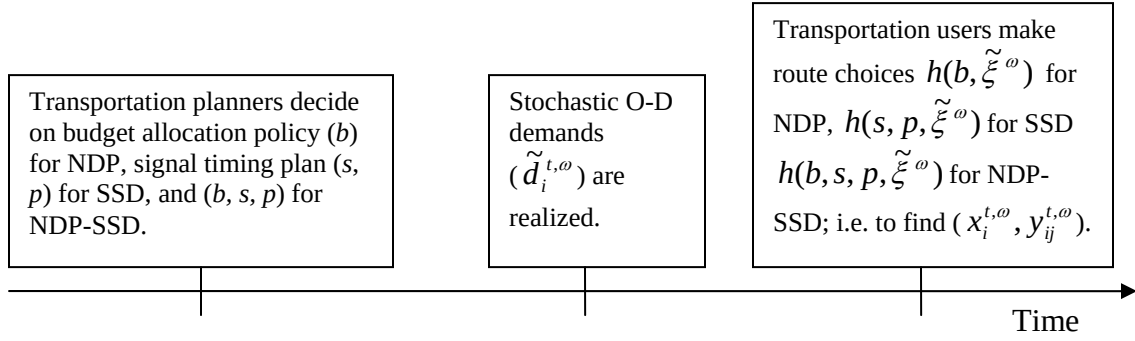


Figure 3.1. Timing of Decisions of Two-Stage Stochastic Models

The recourse function $h(b, \tilde{\xi}^\omega)$ varies depending on the problems as described in the next three sub-sections.

SLP2-SONDP Model

The recourse function of SLP2-SONDP is the SO DTA corresponding to the realized uncertain parameters ($\tilde{\xi}^\omega$) and the assigned budgets (b):

$$h(b, \tilde{\xi}^\omega) = \min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} t \cdot y_{ij}^{t,\omega}$$

subject to $CTM^\omega(x, y, b)$

After substituting this recourse function into the standard SLP2 formulation (3.6), the resulting SLP2-SONDP is (Waller, 2000):

$$\min_{b,x,y} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega}) \quad (3.7.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.7.2)$$

$$CTM^\omega(x, y, b) \quad \forall \omega \in \Omega \quad (3.7.3)$$

SLP2-UONDP Model

The recourse function of SLP2-UONDP is the UO DTA corresponding to the realized uncertain parameters ($\tilde{\xi}^\omega$) and the assigned budgets (b):

$$h(b, \tilde{\xi}^\omega) = \min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega}$$

subject to $CTM^\omega(x, y, b)$

After substituting this recourse function into the standard SLP2 (3.6), the resulting SLP2-UONDP is (Ukkusuri, 2002):

$$\min_{b,x,y} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot M_t \cdot y_{ij}^{t,\omega}) \quad (3.8.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.8.2)$$

$$CTM^\omega(x, y, b) \quad \forall \omega \in \Omega \quad (3.8.3)$$

Like the deterministic case, the common drawback of both SLP2-SONDP and SLP2-UONDP is that these single-level models cannot correctly model the stochastic NDP as the nature of the problem requires the bi-level structure described in the next section.

SLP2-BLPNDP Model

The recourse function of SLP2-BLPNDP is the TSTT calculated from the UODTA corresponding to the realized uncertain parameters ($\tilde{\xi}^\omega$) and the assigned budgets (b):

$$h(b, \tilde{\xi}^\omega) = \sum_{(i,j) \in E_S} \sum_{t \in T} t \cdot y_{ij}^{t,\omega}$$

subject to $\min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega}$

subject to $CTM^\omega(x, y, b)$

After substituting this recourse function into (3.6), the resulting SLP2-BLPNDP is:

$$\min_b \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega}) \quad (3.9.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.9.2)$$

$$\min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega} \quad \forall \omega \in \Omega \quad (3.9.3)$$

$$\text{subject to} \quad CTM^\omega(x, y, b) \quad \forall \omega \in \Omega \quad (3.9.4)$$

It is possible that the solution of the SLP2-BLPNDP yields the least expected cost but with wildly different TSTTs for various O-D demand realizations. This can be better handled by incorporating the higher moments of the distribution of objective value. However, according to the nature of the problem, the higher moments are not considered appropriate, and we adopt a risk term, which is a squared up-side deviation of TSTT from a fixed target TSTT. The rationale is described in the next section.

3.1.3 Robust NDP Model (RO-BLPNDP)

First, we show that the robust NDP has to be bi-level. Let's consider SLP2-SONDP (3.7) and SLP2-UONDP (3.8). The objective functions of these two stochastic programs are minimizing their expected objective values. A naive application of robust optimization could be simply the addition of the risk term. For example, the risk term of SLP2-SONDP could be the variance of TSTT; then, the objective of robust model is

$$\min_{b,x,y} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega}) + \lambda \cdot \sum_{\omega \in \Omega} p^\omega \left(\sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^{t,\omega}) - \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega}) \right)^2$$

where the goal programming weight λ is non-negative.

Clearly, this quadratic program does not have a meaningful implication, since the second-stage decision variables (y_{ij}^t) can play a role to reduce the risk. In other words, transportation users make route choice decisions such that the weighted summation of the expected cost and risk is minimized, which destroy the assumption of SO DTA condition. Therefore, the robust optimization cannot be applied to the single-level models such as SLP2-SONDP and SLP2-UONDP. It only can be applied to the bi-level model SLP2-BLPNDP, since only the first-stage decision variables are used to optimize the objective.

Next, let's consider the SLP2-BLPNDP (3.9). As discussed in Section 2.6, the objective function of the robust optimization model is the tradeoff between the expected cost and risk term, and there are certain possible objective functions as follows (Mulvey et al., 1995):

- $\min_b \bar{z} + \lambda \cdot \sum_{\omega \in \Omega} p^\omega (z^\omega - \bar{z})^2$
- $\min_b \bar{z} + \lambda \cdot \sum_{\omega \in \Omega} p^\omega |z^\omega - \bar{z}|$
- $\min_b \bar{z} + \sum_{\omega \in \Omega} p^\omega (\lambda_u (z^\omega - \bar{z})^+ + \lambda_d (z^\omega - \bar{z})^-)$
- $\min_b \bar{z} + \lambda \cdot \max_{\omega \in \Omega} z^\omega$

where $z^\omega = \sum_{(i,j) \in E_s} \sum_{t \in T} (t \cdot y_{ij}^{t,\omega})$, $\bar{z} = \sum_{\omega \in \Omega} \sum_{(i,j) \in E_s} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega})$, $(z^\omega - \bar{z})^+ = \max(0, z^\omega - \bar{z})$;

$(z^\omega - \bar{z})^- = \max(0, -(z^\omega - \bar{z}))$, λ_u and λ_d are differential weights on up-side and down-side risk.

The last objective function includes the worst-case of risk term, and it may be considered too conservative. For the first three objective functions, it is possible that the optimal solution yields very low risk but very high expected cost. This is indeed not desirable. Thus, the moving target (i.e. $\bar{z}$) in the risk term should not be used, and the fixed target is employed in this study. If transportation planners have the information on the ideal expected TSTT, then it can be set to the target TSTT. Otherwise, in this study, we employ the optimal objective function of the SLP2-BLPNDP as the target TSTT ($TSTT^{TARGET}$). Further, due to the nature of the problem, the up-side and down-side risk should not be considered symmetric. That is, we would not bother and instead prefer if the realized TSTT of a scenario is less than the target TSTT. Thus, the down-side risk is ignored, and only the up-side risk term is taken into account. Specifically, the adopted objective function of the robust model is

$$\min_b \bar{z} + \lambda \cdot \sum_{\omega \in \Omega} p^\omega ((z^\omega - TSTT^{TARGET})^+)^2$$

Formally, the robust optimization model can be viewed as a two-stage stochastic program (3.6). The recourse function of RO-BLPNDP is the tradeoff between the expected TSTT and risk, which are calculated from the UO DTA corresponding to the realized uncertain parameters ($\tilde{\xi}^\omega$) and the assigned budgets (b):

$$\begin{aligned}
h(b, \tilde{\xi}^\omega) &= \min_{D^\omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^{t,\omega}) + \lambda \cdot (D^\omega)^2 \\
\text{subject to} \quad & D^\omega \geq \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^{t,\omega}) - TSTT^{TARGET} \\
& D^\omega \geq 0 \\
& \min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega} \\
& \text{subject to} \quad CTM^\omega(x, y, b)
\end{aligned}$$

It can be noticed that the objective function of the recourse function together with the first two constraints implies that only the up-side deviation from $TSTT^{TARGET}$ is included in the risk term. After substituting this recourse function into (3.6), the resulting continuous quadratic-linear bi-level program of RO-BLPNDP is:

$$\min_{b, D^\omega} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega}) + \lambda \sum_{\omega \in \Omega} (p^\omega \cdot (D^\omega)^2) \quad (3.10.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.10.2)$$

$$D^\omega \geq \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^{t,\omega}) - TSTT^{TARGET} \quad \forall \omega \in \Omega \quad (3.10.3)$$

$$D^\omega \geq 0 \quad \forall \omega \in \Omega \quad (3.10.4)$$

$$\min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega} \quad \forall \omega \in \Omega \quad (3.10.5)$$

$$\text{subject to} \quad CTM^\omega(x, y, b) \quad \forall \omega \in \Omega \quad (3.10.6)$$

It can be noticed that RO-BLPNDP with $\lambda=0$ is essentially the SLP2-BLPNDP. The next section describes an existing SSD model, and the proposed six SSD models.

3.2 Combined DTA and SSD Models

Waller (2000) formulated the deterministic system optimal signal setting design problem (SOSSD), based on the SO DTA LP model, as a mixed-integer program. In this study, we propose six additional models. For stochastic and robust optimization models, only the O-D demands are considered random with known probability distribution. Here,

it is assumed that the possible signal timing plans for all intersections are enumerated, including cycle lengths, green splits, time offsets, and phase sequencing, given fixed signal phasing. The signal coordination is taken into account since time offsets are optimized. We allow different cycle lengths for intersections as opposed to the common cycle length assumption typically found in the literature. The selection of the signal timing plans of all intersections is to be determined by the optimization models. It is noted that this approach assumes a more aggregate model for traffic flow than microscopic (Waller, 2000). Thus, this approach cannot capture certain realities that could impact the optimal signal setting, and it may not yield a better signal timing plan when considering simply an isolated intersection. Rather, these models demonstrate that it is possible to combine network-wide signal setting design with dynamic traffic assignment when accounting for traffic demand uncertainty. The next sub-section describes the three deterministic SSD models.

3.2.1 Deterministic SSD Models (SOSSD, UOSSD and BLPSSD)

Let *INTERSECTION* denote the set of intersection cells. Denote by $P(i, j)$ the set of phases that allow flow from cell i to intersection cell j ; STP_i as the set of signal timing plans for intersection cell i ; P_i as the set of all phases for intersection cell i ; and $phase(i, t)$ as phase number of signal timing plan i at time t . The binary variable s_i equals to 1 if the signal timing plan i is chosen, and 0 otherwise. The variable p_{ij}^t equals to 1 if flow is allowed from cell i to cell j at time t , and 0 otherwise. The three deterministic SSD models (SOSSD, UOSSD and BLPSSD) share the same two sets of constraints: signal and CTM constraints.

First, *SIGNAL_CONSTRAINT*(s, p) represents Constraints (3.11.1)-(3.11.4):

$$\sum_{i \in STP_j} s_i = 1 \quad \forall j \in INTERSECTION \quad (3.11.1)$$

$$p_{ij}^t = \sum_{k \in STP_j: phase(k, t) \in P(i, j)} s_k \quad \forall (i, j) \in RS(j), \forall j \in INTERSECTION, \forall t \in T \quad (3.11.2)$$

$$s_i \in \{0, 1\} \quad \forall i \in \bigcup_{j \in INTERSECTION} STP_j \quad (3.11.3)$$

$$p_{ij}^t \geq 0 \quad \forall (i, j) \in RS(j), \forall j \in INTERSECTION, \forall t \in T \quad (3.11.4)$$

Constraint (3.11.1) enforces that only one signal timing plan is chosen for each intersection. Constraint (3.11.2) determines the variable p_{ij}^t from the summation of signal timing plan variables of intersection cell j that allow flows at time t . This implies that the variables p_{ij}^t need not be constrained to be binary. Constraints (3.11.3) constrained s_i to be binary, and Constraint (3.11.4) are non-negativity constraints of p_{ij}^t .

Second, $CTM(x, y, s, p)$ represents Constraints (3.12.1)-(3.12.11):

$$x_i^t - x_i^{t-1} + \sum_{(i,j) \in FS(i)} y_{ij}^{t-1} - \sum_{(j,i) \in RS(i)} y_{ji}^{t-1} = d_i^t \quad \forall i \in C \setminus C_s, t \in T \quad (3.12.1)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^t - x_i^t \leq 0 \quad \forall i \in C \setminus C_s, t \in T \quad (3.12.2)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^t \leq \delta_i^t (N_i^t - x_i^t) \quad \forall i \in C \setminus (C_R \cup C_s), t \in T \quad (3.12.3)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^t \leq Q_i^t \quad \forall i \in C \setminus (C_R \cup C_s \cup INTERSECTION), t \in T \quad (3.12.4)$$

$$y_{ji}^t \leq p_{ji}^t \cdot Q_i^t \quad \forall (j, i) \in RS(i), i \in INTERSECTION, t \in T \quad (3.12.5)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^t \leq Q_i^t \quad \forall i \in C \setminus C_s, t \in T \quad (3.12.6)$$

$$x_i^0 = \zeta_i \quad \forall i \in C \setminus C_s \quad (3.12.7)$$

$$y_{ij}^0 = 0 \quad \forall (i, j) \in E \quad (3.12.8)$$

$$x_i^{|T|} = 0 \quad \forall i \in C \setminus C_s \quad (3.12.9)$$

$$x_i^t \geq 0 \quad \forall i \in C \setminus C_s, t \in T \quad (3.12.10)$$

$$y_{ij}^t \geq 0 \quad \forall (i, j) \in E, t \in T \quad (3.12.11)$$

The constraint set $CTM(x, y, s, p)$ is the same as the constraints (2.11.2)-(2.11.11) of DTA LP model with the modification, as discussed in Section 2.4.1, to represent the traffic signal at intersection cells (i.e. replace (2.11.5) with (3.12.4)-(3.12.5)). The variables p_{ji}^t are multiplied by the corresponding time-dependent saturation flow Q_i^t to constrain the flows as determined by the signal settings. Since Q_i^t is a parameter, the

cross term ($p_{ji}^t \cdot Q_i^t$) is linear. If a chosen signal timing plan does not allow flow from cell j to cell i at time t , then p_{ij}^t equals to 0, and Constraint (3.12.5) forces y_{ij}^t equal to 0.

For clarity, let us consider Network 2 in Figure 5.12, Chapter 5. There are three intersections ($c3$, $c5$ and $c6$), two origins (cells $c1$ and $c2$), single destination (cell $c7$) and eight analysis period ($|T| = 8$). The pre-determined orderings of signal phases for the three intersections are shown in Figure 5.12f. The pre-determined signal timing plans for intersection $c3$ are shown in Figure 5.13 where ovals represent phase 1, and rectangles represent phase 2. The twelve signal timing plans are labeled $s1$, $s2$, ..., $s12$, respectively. The pre-determined signal timing plans for intersection $c5$ and $c6$ are the same as $c3$. The twelve signal timing plans for intersection $c5$ are labeled $s1001$, $s1002$, ..., $s1012$; and those for $c6$ are labeled $s2001$, $s2002$, ..., $s2012$.

The sets $INTERSECTION$, $P(i,j)$, STP_i , and P_i are as follows:

$$\begin{aligned} INTERSECTION &= \{c3, c5, c6\}; STP_{c3} = \{s1, s2, \dots, s12\}; STP_{c5} = \{s1001, s1002, \dots, s1012\}; \\ STP_{c6} &= \{s2001, s2002, \dots, s2012\}; P_{c3} = P_{c5} = P_{c6} = \{phase1, phase2\}; P(c1, c3) = \{phase1\}; \\ P(c2, c3) &= \{phase2\}; P(c2, c5) = \{phase1\}; P(c1, c5) = \{phase2\}; P(c5, c6) = \{phase1\}; \\ P(c3, c6) &= \{phase2\}. \end{aligned}$$

The parameters $phase(i,t)$ are shown below:

$$\begin{aligned} phase(s1, t) &= phase1, \text{ for } t = t1, t3, t5, t7 \\ phase(s1, t) &= phase2, \text{ for } t = t2, t4, t6, t8 \\ phase(s2, t) &= phase1, \text{ for } t = t2, t4, t6, t8 \\ phase(s2, t) &= phase2, \text{ for } t = t1, t3, t5, t7 \\ &\dots \\ phase(s12, t) &= phase1, \text{ for } t = t1, t4, t5, t8 \\ phase(s12, t) &= phase2, \text{ for } t = t2, t3, t6, t7 \end{aligned}$$

The binary decision variables $s_i \in \{0,1\} \forall i \in \cup_{j \in INTERSECTION} STP_j$ are:

$$\begin{aligned} s_1, s_2, \dots, s_{12} &\text{ for intersection } c3 \\ s_{1001}, s_{1002}, \dots, s_{1012} &\text{ for intersection } c5 \\ s_{2001}, s_{2002}, \dots, s_{2012} &\text{ for intersection } c6 \end{aligned}$$

Constraint (3.11.2) can be specified for the three intersections as follows:

Intersection c3:

$$p_{13}^{t1} = s_1 + s_3 + s_7 + s_8 + s_9 + s_{12}, \quad p_{23}^{t1} = s_2 + s_4 + s_5 + s_6 + s_{10} + s_{11},$$

$$p_{13}^{t2} = s_2 + s_4 + s_6 + s_8 + s_9 + s_{10}, \quad p_{23}^{t2} = s_1 + s_3 + s_5 + s_7 + s_{11} + s_{12},$$

...

$$p_{13}^{t8} = s_2 + s_4 + s_6 + s_8 + s_{11} + s_{12}, \quad p_{23}^{t8} = s_1 + s_3 + s_5 + s_7 + s_9 + s_{10}$$

Intersection c5:

$$p_{25}^{t1} = s_{1001} + s_{1003} + s_{1007} + s_{1008} + s_{1009} + s_{1012}, \quad p_{15}^{t1} = s_{1002} + s_{1004} + s_{1005} + s_{1006} + s_{1010} + s_{1011},$$

$$p_{25}^{t2} = s_{1002} + s_{1004} + s_{1006} + s_{1008} + s_{1009} + s_{1010}, \quad p_{15}^{t2} = s_{1001} + s_{1003} + s_{1005} + s_{1007} + s_{1011} + s_{1012},$$

...

$$p_{25}^{t8} = s_{1002} + s_{1004} + s_{1006} + s_{1008} + s_{1011} + s_{1012}, \quad p_{15}^{t8} = s_{1001} + s_{1003} + s_{1005} + s_{1007} + s_{1009} + s_{1010}$$

Intersection c6:

$$p_{56}^{t1} = s_{2001} + s_{2003} + s_{2007} + s_{2008} + s_{2009} + s_{2012}, \quad p_{36}^{t1} = s_{2002} + s_{2004} + s_{2005} + s_{2006} + s_{2010} + s_{2011}$$

$$p_{56}^{t2} = s_{2002} + s_{2004} + s_{2006} + s_{2008} + s_{2009} + s_{2010}, \quad p_{36}^{t2} = s_{2001} + s_{2003} + s_{2005} + s_{2007} + s_{2011} + s_{2012},$$

...

$$p_{56}^{t8} = s_{2002} + s_{2004} + s_{2006} + s_{2008} + s_{2011} + s_{2012}, \quad p_{36}^{t8} = s_{2001} + s_{2003} + s_{2005} + s_{2007} + s_{2009} + s_{2010}$$

The next three sub-sections describe three deterministic SSD models.

SOSSD Model

The objective of the SOSSD is to determine the optimal signal setting so as to minimize the TSTT under SO DTA behavior. The mixed-inter linear program (MIP) of SOSSD is (Waller, 2000):

$$\min_{s, p, x, y} \sum_{(i, j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^t) \quad (3.13.1)$$

$$\text{subject to} \quad \text{SIGNAL_CONSTRAINT}(s, p) \quad (3.13.2)$$

$$\text{CTM}(x, y, s, p) \quad (3.13.3)$$

UOSSD Model

The objective of the UOSSD is to determine the optimal signal setting such that the individual users minimize their own travel time. It is noted that the TSTT is not optimized. We extend SOSSD to UOSSD in the same fashion as UONDP was extended from SONDP. The UOSSD model is:

$$\min_{s,p,x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} (M_t \cdot y_{ij}^t) \quad (3.14.1)$$

$$\text{subject to} \quad \text{SIGNAL_CONSTRAINT}(s, p) \quad (3.14.2)$$

$$\text{CTM}(x, y, s, p) \quad (3.14.3)$$

The common drawback of SOSSD and UOSSD is that they do not model the problem naturally, since by nature the SSD problem is bi-level. The next section describes the improved bi-level SSD model.

BLPSSD Model

The objective of the BLPSSD is to determine the optimal signal setting such that the TSTT is minimized under UO DTA behavior. The mixed-zero-one continuous linear bi-level program of BLPSSD is:

$$\min_{s,p} \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^t) \quad (3.15.1)$$

$$\text{subject to} \quad \text{SIGNAL_CONSTRAINT}(s, p) \quad (3.15.2)$$

$$\min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} (M_t \cdot y_{ij}^t) \quad (3.15.3)$$

$$\text{subject to} \quad \text{CTM}(x, y, s, p) \quad (3.15.4)$$

The three deterministic SSD models assume all parameters are deterministic. These are unrealistic as the parameters, especially O-D demands, are stochastic. The better approach to model the SSD is the two-stage stochastic programming described in the next section.

3.2.2 Two-Stage Stochastic SSD Models (SLP2-SOSSD, SLP2-UOSSD and SLP2-BLPSSD)

The three two-stage stochastic linear programs (SLP2-SOSSD, SLP2-UOSSD and SLP2-BLPSSD) and the robust model (RO-BLPSSD) employ the signal constraints (3.11.1)-(3.11.4) and the following CTM constraints corresponding to each realization of stochastic O-D demands.

Let $CTM^\omega(x, y, s, p)$, $\forall \omega \in \Omega$ represent Constraints (3.16.1)-(3.16.11):

$$x_i^{t,\omega} - x_i^{t-1,\omega} + \sum_{(i,j) \in FS(i)} y_{ij}^{t-1,\omega} - \sum_{(j,i) \in RS(i)} y_{ji}^{t-1,\omega} = \tilde{d}_i^{t,\omega} \quad \forall i \in C \setminus C_s, t \in T \quad (3.16.1)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^{t,\omega} - x_i^{t,\omega} \leq 0 \quad \forall i \in C \setminus C_s, t \in T \quad (3.16.2)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^{t,\omega} \leq \delta_i^t (N_i^t - x_i^{t,\omega}) \quad \forall i \in C \setminus (C_R \cup C_s), t \in T \quad (3.16.3)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^{t,\omega} \leq Q_i^t \quad \forall i \in C \setminus (C_R \cup C_s \cup INTERSECTION), t \in T \quad (3.16.4)$$

$$y_{ji}^{t,\omega} \leq p_{ji}^t \cdot Q_i^t \quad \forall (j,i) \in RS(i), i \in INTERSECTION, t \in T \quad (3.16.5)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^{t,\omega} \leq Q_i^t \quad \forall i \in C \setminus C_s, t \in T \quad (3.16.6)$$

$$x_i^{0,\omega} = \zeta_i \quad \forall i \in C \setminus C_s \quad (3.16.7)$$

$$y_{ij}^{0,\omega} = 0 \quad \forall (i,j) \in E \quad (3.16.8)$$

$$x_i^{[T],\omega} = 0 \quad \forall i \in C \setminus C_s \quad (3.16.9)$$

$$x_i^{t,\omega} \geq 0 \quad \forall i \in C \setminus C_s, t \in T \quad (3.16.10)$$

$$y_{ij}^{t,\omega} \geq 0 \quad \forall (i,j) \in E, t \in T \quad (3.16.11)$$

The standard two-stage stochastic signal setting design model can be written as:

$$\min_{s,p} E[h(s, p, \tilde{\xi})] = \sum_{\omega \in \Omega} [p^\omega \cdot h(s, p, \tilde{\xi}^\omega)] \quad (3.17.1)$$

$$\text{subject to} \quad \text{SIGNAL_CONSTRAINT}(s, p) \quad (3.17.2)$$

The timing of decisions is similar to NDP shown in Figure 3.1. The recourse function $h(s, p, \tilde{\xi}^\omega)$ depends on the problems as described in the next three sub-sections.

SLP2-SOSSD Model

The recourse function of SLP2-SOSSD is:

$$h(s, p, \tilde{\xi}^\omega) = \min_{x, y} \sum_{(i, j) \in E_S} \sum_{t \in T} t \cdot y_{ij}^{t, \omega}$$

subject to $CTM^\omega(x, y, s, p)$

After substituting this recourse function into (3.17), the resulting SLP2-SOSSD is:

$$\min_{s, p, x, y} \sum_{\omega \in \Omega} \sum_{(i, j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t, \omega}) \quad (3.18.1)$$

$$\text{subject to } \quad \text{SIGNAL_CONSTRAINT}(s, p) \quad (3.18.2)$$

$$CTM^\omega(x, y, s, p) \quad \forall \omega \in \Omega \quad (3.18.3)$$

SLP2-UOSSD Model

The recourse function of SLP2-UOSSD is:

$$h(s, p, \tilde{\xi}^\omega) = \min_{x, y} \sum_{(i, j) \in E_S} \sum_{t \in T} M_t \cdot y_{ij}^{t, \omega}$$

subject to $CTM^\omega(x, y, s, p)$

Substituting this recourse function into (3.17) yields SLP2-UOSSD:

$$\min_{s, p, x, y} \sum_{\omega \in \Omega} \sum_{(i, j) \in E_S} \sum_{t \in T} (p^\omega \cdot M_t \cdot y_{ij}^{t, \omega}) \quad (3.19.1)$$

$$\text{subject to } \quad \text{SIGNAL_CONSTRAINT}(s, p) \quad (3.19.2)$$

$$CTM^\omega(x, y, s, p) \quad \forall \omega \in \Omega \quad (3.19.3)$$

Since the nature of the stochastic SSD is bi-level, the preferred bi-level formulation is presented next.

SLP2-BLPSSD Model

The recourse function of SLP2-BLPSSD is:

$$\begin{aligned}
 h(s, p, \tilde{\xi}^\omega) &= \sum_{(i,j) \in E_S} \sum_{t \in T} t \cdot y_{ij}^{t,\omega} \\
 \text{subject to} \quad & \min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega} \\
 & \text{subject to} \quad CTM^\omega(x, y, s, p)
 \end{aligned}$$

After substituting this recourse function into (3.17), the resulting mixed-zero-one continuous linear bi-level model (SLP2-BLPSSD) is:

$$\min_{s,p} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega}) \quad (3.20.1)$$

$$\text{subject to} \quad SIGNAL_CONSTRAINT(s, p) \quad (3.20.2)$$

$$\min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega} \quad \forall \omega \in \Omega \quad (3.20.3)$$

$$\text{subject to} \quad CTM^\omega(x, y, s, p) \quad \forall \omega \in \Omega \quad (3.20.4)$$

It is possible that the solution of the SLP2-BLPSSD yields the least expected TSTT but with drastically high expected risk. This can be better handled by modeling the stochastic SSD as a robust optimization model as described in the next section.

3.2.3 Robust SSD Model (RO-BLPSSD)

The discussion for the robust NDP model in Section 3.1.3 can also be applied here. Essentially, the robust SSD model has to be bi-level in order to be meaningful. The risk term with a moving target TSTT (i.e. the expected TSTT) in the objective function as well as the worst-case risk term is considered undesirable. We consider the objective that penalizes only the up-side deviation from the fixed target cost. The proposed robust SSD model is a two-stage stochastic program, and the recourse function of RO-BLPSSD is:

$$\begin{aligned}
h(s, p, \tilde{\xi}^\omega) &= \min_{D^\omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^{t,\omega}) + \lambda \cdot (D^\omega)^2 \\
\text{subject to} \quad & D^\omega \geq \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^{t,\omega}) - TSTT^{TARGET} \\
& D^\omega \geq 0 \\
& \min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega} \\
& \text{subject to} \quad CTM^\omega(x, y, s, p)
\end{aligned}$$

It can be noticed that the objective function of the recourse function together with the first two constraints implies that only the up-side deviation from $TSTT^{TARGET}$ is penalized. Substituting this recourse function into the standard two-stage stochastic SSD model (3.17) yields a mixed-zero-one quadratic-linear bilevel model (RO-BLPSSD):

$$\min_{s,p,D^\omega} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega}) + \lambda \sum_{\omega \in \Omega} (p^\omega \cdot (D^\omega)^2) \quad (3.21.1)$$

$$\text{subject to} \quad SIGNAL_CONSTRAINT(s, p) \quad (3.21.2)$$

$$D^\omega \geq \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^{t,\omega}) - TSTT^{TARGET} \quad \forall \omega \in \Omega \quad (3.21.3)$$

$$D^\omega \geq 0 \quad \forall \omega \in \Omega \quad (3.21.4)$$

$$\min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega} \quad \forall \omega \in \Omega \quad (3.21.5)$$

$$\text{subject to} \quad CTM^\omega(x, y, s, p) \quad \forall \omega \in \Omega \quad (3.21.6)$$

The next section describes an existing NDP-SSD model and the proposed six NDP-SSD models.

3.3 Integrated NDP, SSD and DTA Problem (NDP-SSD)

Waller (2000) formulated the combined NDP, SSD and SO DTA model (SONDP-SSD) based on the SO DTA LP model. In this study, we propose six additional NDP-SSD models. For stochastic and robust optimization models, only the O-D demands are considered random with known probability distribution. The assumptions of NDP and

SSD also apply to the combined problem (NDP-SSD). The next section describes deterministic NDP-SSD models.

3.3.1 Deterministic NDP-SSD Models (SONDP-SSD, UONDP-SSD and BLPNDP-SSD)

The three deterministic NDP-SSD models (SONDP-SSD, UONDP-SSD and BLPNDP-SSD) employ the same three sets of constraints: the budget constraints (3.1.1)-(3.1.2), the signal constraints (3.11.1)-(3.11.4) and the following CTM constraints:

Let $CTM(x, y, b, s, p)$ represent Constraints (3.22.1)-(3.22.12):

$$x_i^t - x_i^{t-1} + \sum_{(i,j) \in FS(i)} y_{ij}^{t-1} - \sum_{(j,i) \in RS(i)} y_{ji}^{t-1} = d_i^t \quad \forall i \in C \setminus C_s, t \in T \quad (3.22.1)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^t - x_i^t \leq 0 \quad \forall i \in C \setminus C_s, t \in T \quad (3.22.2)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^t \leq \delta_i^t (N_i^t + \chi_i \cdot b_i - x_i^t) \quad \forall i \in C \setminus (C_R \cup C_s), t \in T \quad (3.22.3)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^t \leq Q_i^t + \phi_i \cdot b_i \quad \forall i \in C \setminus (C_R \cup C_s \cup INTERSECTION), t \in T \quad (3.22.4)$$

$$y_{ji}^t \leq Q_i^t + \phi_i \cdot b_i \quad \forall (j,i) \in RS(i), i \in INTERSECTION, t \in T \quad (3.22.5)$$

$$y_{ji}^t \leq p_{ji}^t (Q_i^t + \phi_i \cdot TAB) \quad \forall (j,i) \in RS(i), i \in INTERSECTION, t \in T \quad (3.22.6)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^t \leq Q_i^t + \phi_i \cdot b_i \quad \forall i \in C \setminus C_s, t \in T \quad (3.22.7)$$

$$x_i^0 = \zeta_i \quad \forall i \in C \setminus C_s \quad (3.22.8)$$

$$y_{ij}^0 = 0 \quad \forall (i,j) \in E \quad (3.22.9)$$

$$x_i^{|T|} = 0 \quad \forall i \in C \setminus C_s \quad (3.22.10)$$

$$x_i^t \geq 0 \quad \forall i \in C \setminus C_s, t \in T \quad (3.22.11)$$

$$y_{ij}^t \geq 0 \quad \forall (i,j) \in E, t \in T \quad (3.22.12)$$

These CTM constraints are the DTA constraints when accounting for the network improvement policies (b_i) and the signal settings (p_{ij}^t). If $p_{ji}^t = 1$, y_{ji}^t is constrained by the

minimum of $Q_i^t + \phi_i \cdot b_i$ and $Q_i^t + \phi_i \cdot TAB$. Clearly, $Q_i^t + \phi_i \cdot TAB \geq Q_i^t + \phi_i \cdot b_i$. Hence, if $p_{ji}^t = 1$, y_{ji}^t is constrained by $Q_i^t + \phi_i \cdot b_i$. If a chosen signal timing plan does not allow flow from cell j to cell i at time t ($p_{ji}^t = 0$), then y_{ji}^t is constrained to be zero. It is noted that Constraints (3.22.5)-(3.22.6) are a linearized set of constraints to the initial non-linear constraint $y_{ji}^t \leq p_{ji}^t (Q_i^t + \phi_i \cdot b_i) \quad \forall (j, i) \in RS(i), i \in INTERSECTION, t \in T$. The next three sub-sections describe three deterministic NDP-SSD models.

SONDP-SSD Model

The objective of the SONDP-SSD is to simultaneously determine the optimal network improvement policy and signal setting so as to minimize the TSTT under SO DTA behavior. The MIP of SONDP-SSD is (Waller, 2000):

$$\min_{b, s, p, x, y} \sum_{(i, j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^t) \quad (3.23.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.23.2)$$

$$SIGNAL_CONSTRAINT(s, p) \quad (3.23.3)$$

$$CTM(x, y, b, s, p) \quad (3.23.4)$$

UONDP-SSD Model

The objective of the UONDP-SSD is to simultaneously determine the optimal network improvement policy and signal setting such that the individual users minimize their own travel time. It is noted that the TSTT is not optimized. The MIP of UONDP-SSD is:

$$\min_{b, s, p, x, y} \sum_{(i, j) \in E_S} \sum_{t \in T} (M_t \cdot y_{ij}^t) \quad (3.24.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.24.2)$$

$$SIGNAL_CONSTRAINT(s, p) \quad (3.24.3)$$

$$CTM(x, y, b, s, p) \quad (3.24.4)$$

The next section describes the more realistic bi-level NDP-SSD model.

BLPNDP-SSD Model

The objective of the BLPNDP-SSD is to simultaneously determine the optimal network improvement policy and signal setting such that the TSTT is minimized under UO DTA behavior. The mixed-zero-one continuous linear bi-level model (BLPNDP-SSD) is:

$$\min_{b,s,p} \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^t) \quad (3.25.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.25.2)$$

$$SIGNAL_CONSTRAINT(s, p) \quad (3.25.3)$$

$$\min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} (M_t \cdot y_{ij}^t) \quad (3.25.4)$$

$$\text{subject to} \quad CTM(x, y, b, s, p) \quad (3.25.5)$$

The next section describes the stochastic NDP-SSD models that account for uncertain O-D demands.

3.3.2 Two-Stage Stochastic NDP-SSD Models (SLP2-SONDP-SSD, SLP2-UONDP-SSD and SLP2-BLPNDP-SSD)

The three two-stage stochastic linear programs (SLP2-SONDP-SSD, SLP2-UONDP-SSD and SLP2-BLPNDP-SSD) and the robust model (RO-BLPNDP-SSD) employ the budget constraints (3.1.1)-(3.1.2), the signal constraints (3.11.1)-(3.11.4) and the following CTM constraints corresponding to each realization of stochastic O-D demands.

Let $CTM^\omega(x, y, b, s, p)$, $\forall \omega \in \Omega$ represent Constraints (3.26.1)-(3.26.12):

$$x_i^{t,\omega} - x_i^{t-1,\omega} + \sum_{(i,j) \in FS(i)} y_{ij}^{t-1,\omega} - \sum_{(j,i) \in RS(i)} y_{ji}^{t-1,\omega} = \tilde{d}_i^{t,\omega} \quad \forall i \in C \setminus C_s, t \in T \quad (3.26.1)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^{t,\omega} - x_i^{t,\omega} \leq 0 \quad \forall i \in C \setminus C_s, t \in T \quad (3.26.2)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^{t,\omega} \leq \delta_i^t (N_i^t + \chi_i \cdot b_i - x_i^{t,\omega}) \quad \forall i \in C \setminus (C_R \cup C_s), t \in T \quad (3.26.3)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^{t,\omega} \leq Q_i^t + \phi_i \cdot b_i \quad \forall i \in C \setminus (C_R \cup C_S \cup INTERSECTION), t \in T \quad (3.26.4)$$

$$y_{ji}^{t,\omega} \leq Q_i^t + \phi_i \cdot b_i \quad \forall (j,i) \in RS(i), i \in INTERSECTION, t \in T \quad (3.26.5)$$

$$y_{ji}^{t,\omega} \leq p_{ji}^t (Q_i^t + \phi_i \cdot TAB) \quad \forall (j,i) \in RS(i), i \in INTERSECTION, t \in T \quad (3.26.6)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^{t,\omega} \leq Q_i^t + \phi_i \cdot b_i \quad \forall i \in C \setminus C_S, t \in T \quad (3.26.7)$$

$$x_i^{0,\omega} = \zeta_i \quad \forall i \in C \setminus C_S \quad (3.26.8)$$

$$y_{ij}^{0,\omega} = 0 \quad \forall (i,j) \in E \quad (3.26.9)$$

$$x_i^{[T],\omega} = 0 \quad \forall i \in C \setminus C_S \quad (3.26.10)$$

$$x_i^{t,\omega} \geq 0 \quad \forall i \in C \setminus C_S, t \in T \quad (3.26.11)$$

$$y_{ij}^{t,\omega} \geq 0 \quad \forall (i,j) \in E, t \in T \quad (3.26.12)$$

The standard two-stage stochastic program for NDP-SSD can be written as:

$$\min_{b,s,p} E[h(b,s,p,\tilde{\xi})] = \sum_{\omega \in \Omega} [p^\omega \cdot h(b,s,p,\tilde{\xi}^\omega)] \quad (3.27.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.27.2)$$

$$SIGNAL_CONSTRAINT(s,p) \quad (3.27.3)$$

The timing of decisions is similar to NDP and SSD as shown in Figure 3.1. The recourse function $h(b,s,p,\tilde{\xi}^\omega)$ depends on the problems as described in the next three sub-sections.

SLP2-SONDP-SSD Model

The recourse function of SLP2-SONDP-SSD is:

$$h(b,s,p,\tilde{\xi}^\omega) = \min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} t \cdot y_{ij}^{t,\omega}$$

$$\text{subject to} \quad CTM^\omega(x,y,b,s,p)$$

Substituting this recourse function into (3.27) yields the MIP of SLP2-SONDP-SSD:

$$\min_{b,s,p,x,y} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega}) \quad (3.28.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.28.2)$$

$$SIGNAL_CONSTRAINT(s, p) \quad (3.28.3)$$

$$CTM^\omega(x, y, b, s, p) \quad \forall \omega \in \Omega \quad (3.28.4)$$

SLP2-UONDP-SSD Model

The recourse function of SLP2-UONDP-SSD is:

$$h(b, s, p, \tilde{\xi}^\omega) = \min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t \cdot y_{ij}^{t,\omega}$$

$$\text{subject to} \quad CTM^\omega(x, y, b, s, p)$$

Substituting this recourse function into (3.27) yields the MIP of SLP2-UONDP-SSD:

$$\min_{b,s,p,x,y} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot M_t \cdot y_{ij}^{t,\omega}) \quad (3.29.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.29.2)$$

$$SIGNAL_CONSTRAINT(s, p) \quad (3.29.3)$$

$$CTM^\omega(x, y, b, s, p) \quad \forall \omega \in \Omega \quad (3.29.4)$$

The next section describes the preferred bi-level stochastic model.

SLP2-BLPNDP-SSD Model

The recourse function of SLP2-BLPNDP-SSD is:

$$h(b, s, p, \tilde{\xi}^\omega) = \sum_{(i,j) \in E_S} \sum_{t \in T} t \cdot y_{ij}^{t,\omega}$$

$$\text{subject to} \quad \min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega}$$

$$\text{subject to} \quad CTM^\omega(x, y, b, s, p)$$

Substituting this recourse function into (3.27) yields a mixed-zero-one continuous linear bi-level model (SLP2-BLPNDP-SSD):

$$\min_{b,s,p} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega}) \quad (3.30.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.30.2)$$

$$SIGNAL_CONSTRAINT(s, p) \quad (3.30.3)$$

$$\min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega} \quad \forall \omega \in \Omega \quad (3.30.4)$$

$$\text{subject to} \quad CTM^\omega(x, y, b, s, p) \quad \forall \omega \in \Omega \quad (3.30.5)$$

The next section describes the robustness approach for NDP-SSD to deal with the O-D demand uncertainty.

3.3.3 Robust NDP-SSD Model (RO-BLPNDP-SSD)

The discussion for the robust NDP in Sections 3.1.3 can be applied here. Basically, the robust NDP-SSD model has to be bi-level, and we choose the risk term such that only the up-side deviation from the target TSTT is penalized. RO-BLPNDP-SSD can be considered as a two-stage stochastic program, and the recourse function of RO-BLPNDP-SSD is:

$$\begin{aligned} h(b, s, p, \tilde{\xi}^\omega) &= \min_{D^\omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^{t,\omega}) + \lambda \cdot (D^\omega)^2 \\ \text{subject to} \quad D^\omega &\geq \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^{t,\omega}) - TSTT^{TARGET} \\ D^\omega &\geq 0 \\ \min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega} \\ \text{subject to} \quad &CTM^\omega(x, y, b, s, p) \end{aligned}$$

It can be noticed that the objective function of the recourse function together with the first two constraints implies that only the up-side deviation from $TSTT^{TARGET}$ is included in the risk term. Substituting this recourse function into (3.27) yields a mixed-zero-one continuous quadratic-linear bi-level model (RO-BLPNDP-SSD):

$$\min_{b,s,p,D^\omega} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega}) + \lambda \sum_{\omega \in \Omega} (p^\omega \cdot (D^\omega)^2) \quad (3.31.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.31.2)$$

$$SIGNAL_CONSTRAINT(s, p) \quad (3.31.3)$$

$$D^\omega \geq \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^{t,\omega}) - TSTT^{TARGET} \quad \forall \omega \in \Omega \quad (3.31.4)$$

$$D^\omega \geq 0 \quad \forall \omega \in \Omega \quad (3.31.5)$$

$$\min_{x,y} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega} \quad \forall \omega \in \Omega \quad (3.31.6)$$

$$\text{subject to} \quad CTM^\omega(x, y, b, s, p) \quad \forall \omega \in \Omega \quad (3.31.7)$$

The next section describes the exact solution methods for the NDP models.

3.4 Exact Solution Methods for NDP Models

The single-level models (SONDP, UONDP, SLP2-SONDP and SLP2-UONDP) are linear programs, and can be directly solved by any LP solver (CPLEX in this study). The solution methods for bi-level programming models (BLPNDP, SLP2-BLPNDP and RO-BLPNDP) in Section 3.1 are described, respectively.

3.4.1 Exact Solution Methods for BLPNDP

This study employs two methods that belong to the first two categories of viable algorithms as discussed in Section 2.7. The first method, a vertex-enumeration-type algorithm, is the K^{th} -best algorithm originally invented by Murty (1968) and coined the name by Bialas and Karwan (1982). We adopt this algorithm because it solves a series of the known subproblem UO DTA, and this potentially leads to a heuristic. Moreover, the advantage of this over the others is that if storage or computational limits are reached before convergence, the algorithm can return the best solution found with an optimality gap. The other methods yield a feasible and optimal solution only at the end of the algorithm. The second approach based on the KKT conditions is to reformulate the problem as a mixed-integer program (MIP), proposed by Fortuny-Amat and McCarl (1981). This approach is employed because it is relatively easy to implement, and it can

provide the exact solution to double check with the first approach. The two methods are described as follows.

3.4.1.1 K^{th} -Best Algorithm

Let us consider the standard form of L-BLPP for convenient discussion:

$$\min_{x \in X} F(x, y) = c_1 x + d_1 y \quad (3.32.1)$$

$$\text{subject to} \quad A_1 x + B_1 y \leq b_1 \quad (3.32.2)$$

$$\min_{y \in Y} f(x, y) = c_2 x + d_2 y \quad (3.32.3)$$

$$\text{subject to} \quad A_2 x + B_2 y \leq b_2 \quad (3.32.4)$$

where $x \in X \subset R^n$, $y \in Y \subset R^m$, $F : X \times Y \rightarrow R^1$, $f : X \times Y \rightarrow R^1$, $c_1, c_2 \in R^n$,

$d_1, d_2 \in R^m$, $b_1 \in R^p$, $b_2 \in R^q$, $A_1 \in R^{p \times n}$, $B_1 \in R^{p \times m}$, $A_2 \in R^{q \times n}$, $B_2 \in R^{q \times m}$

$X = \{x : x \geq 0\}$ and $Y = \{y : y \geq 0\}$.

The K^{th} -best algorithm makes use of Facts 1-4 in Section 2.7. This algorithm involves the extreme point ranking procedure in the context of the simplex method. In the standard form, let $(x_{[1]}, y_{[1]}), (x_{[2]}, y_{[2]}), \dots, (x_{[N]}, y_{[N]})$ denote the N ordered basic feasible solutions (BFS) to the following linear program:

$$\min\{F(x, y) = c_1 x + d_1 y : (x, y) \in S\} \quad (3.33)$$

such that $F(x_{[1]}, y_{[1]}) \leq F(x_{[2]}, y_{[2]}) \leq \dots \leq F(x_{[N]}, y_{[N]})$. Then, solving the L-BLPP (3.32) is equivalent to finding the index $K^* = \min\{i \in \{1, \dots, N\} : (x_{[i]}, y_{[i]}) \in IR\}$ with the global optimum of $(x_{[K^*]}, y_{[K^*]})$ (i.e. the $(K^*)^{\text{th}}$ best extreme point solution to (3.33)). Note that there are three assumptions for this solution method. First, S is nonempty and compact. Second, $P(x)$ is nonempty. Third, $P(x)$ is single-valued for all permissible x . The algorithm is described as follows (Bard, 1998):

Step 1: Set $i=1$. Solve problem (3.33) with the simplex method to obtain the optimal solution $(x_{[1]}, y_{[1]})$. Let $W = \{(x_{[1]}, y_{[1]})\}$ and $T = \{\}$. Go to Step 2.

Step 2: Solve the follower's linear program (3.34) to see if the current point is in the rational reaction set $P(x_{[i]})$.

$$\min\{d_2 y : y \in P(x_{[i]})\} \quad (3.34)$$

where $P(x_{[i]}) = \{y \in Y : y \in \arg \min[f(x_{[i]}, \hat{y}) : \hat{y} \in S(x_{[i]})]\}$ and $S(x_{[i]}) = \{y \in Y : B_2 y \leq b_2 - A_2 x_{[i]}\}$. Let $\tilde{y}$ denote the optimal solution to (3.34). If $\tilde{y} = y_{[i]}$, stop; $(x_{[i]}, y_{[i]})$ is the global optimum to problem (3.32) with $K^* = i$. Otherwise, go to Step 3.

Step 3: Let $W_{[i]}$ denote the set of adjacent extreme points of $(x_{[i]}, y_{[i]})$ such that

$(x, y) \in W_{[i]}$ implies $c_1 x + d_1 y \geq c_1 x_{[i]} + d_1 y_{[i]}$. Let $T = T \cup \{(x_{[i]}, y_{[i]})\}$ and $W = (W \cup W_{[i]}) \setminus T$. Go to Step 4.

Step 4: Set $i = i+1$ and choose $(x_{[i]}, y_{[i]})$ so that $c_1 x_{[i]} + d_1 y_{[i]} =$

$\min\{c_1 x + d_1 y : (x, y) \in W\}$. Go to Step 2.

Subsequently, we further describe the K^{th} best algorithm in the context of the original problem (3.4) because it provides some intuition for possible heuristic. Also, for better understanding, we illustrate the algorithm using Network 1 in Figure 5.1 in Chapter 5 with deterministic time-dependent demands in Figure 5.2. For Network 1, there are 149 independent constraints in SONDP, resulting in 149 basic variables. There are five upper-level decision variables (i.e. budget assigned to cells 1-5) and 189 lower-level decision variables, resulting in the total of 194 variables.

The data structure used in this study for Network 1 is shown in Figure 3.2. The variables LB and UB are the lower and upper bounds, respectively. The structure array $Array_list[il]$ stores the il^{th} generated basic feasible solutions (BFSs), and the structure array $Array_rank[ir]$ stores the ir^{th} ranked BFSs. The variables SO_TSTT and UO_TSTT are the TSTT under SODTA and UO DTA behaviors, respectively. The integer array

$BVI[149]$ stores the indices of basic variables. The double array $budget[5]$ stores the value of budget variables. The matrix $Simplex_tableau[150][195]$ is the simplex tableau of the SONDP.

double <i>LB</i> double <i>UB</i> struct <i>*Array_list</i> <table><tr><td>Double <i>SO_TSTT</i></td><td>...</td><td>...</td><td>...</td></tr><tr><td>int <i>BVI</i>[149]</td><td></td><td></td><td></td></tr><tr><td>int <i>picked</i></td><td></td><td></td><td></td></tr></table> struct <i>*Array_rank</i> <table><tr><td>double <i>SO_TSTT</i></td><td>...</td><td>...</td><td>...</td></tr><tr><td>double <i>UO_TSTT</i></td><td></td><td></td><td></td></tr><tr><td>int <i>BVI</i>[149]</td><td></td><td></td><td></td></tr><tr><td>double <i>budget</i>[5]</td><td></td><td></td><td></td></tr></table>				Double <i>SO_TSTT</i>	...	...	...	int <i>BVI</i> [149]				int <i>picked</i>				double <i>SO_TSTT</i>	...	...	...	double <i>UO_TSTT</i>				int <i>BVI</i> [149]				double <i>budget</i> [5]				Double <i>Simplex_tableau</i> [150][195] <table><tr><td><i>cc</i>₁</td><td><i>cc</i>₂</td><td>...</td><td><i>cc</i>_{<i>j</i>}</td><td>...</td><td><i>cc</i>_{<i>n</i>}</td><td><i>Z</i></td></tr><tr><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td><td><i>bb</i>₁</td></tr><tr><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td><td><i>bb</i>₂</td></tr><tr><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td></tr><tr><td><i>AA</i>₁</td><td><i>AA</i>₂</td><td>.</td><td><i>AA</i>_{<i>j</i>}</td><td>.</td><td><i>AA</i>_{<i>n</i>}</td><td>.</td></tr><tr><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td><td><i>bb</i>_{<i>i</i>}</td></tr><tr><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td></tr><tr><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td><td><i>bb</i>_{<i>m</i>}</td></tr></table>							<i>cc</i> ₁	<i>cc</i> ₂	...	<i>cc</i> _{<i>j</i>}	...	<i>cc</i> _{<i>n</i>}	<i>Z</i>	.	.	.	.	.	.	<i>bb</i> ₁	.	.	.	.	.	.	<i>bb</i> ₂	.	.	.	.	.	.	.	<i>AA</i> ₁	<i>AA</i> ₂	.	<i>AA</i> _{<i>j</i>}	.	<i>AA</i> _{<i>n</i>}	.	.	.	.	.	.	.	<i>bb</i> _{<i>i</i>}	.	.	.	.	.	.	.	.	.	.	.	.	.	<i>bb</i> _{<i>m</i>}
Double <i>SO_TSTT</i>	...	...	...																																																																																											
int <i>BVI</i> [149]																																																																																														
int <i>picked</i>																																																																																														
double <i>SO_TSTT</i>	...	...	...																																																																																											
double <i>UO_TSTT</i>																																																																																														
int <i>BVI</i> [149]																																																																																														
double <i>budget</i> [5]																																																																																														
<i>cc</i> ₁	<i>cc</i> ₂	...	<i>cc</i> _{<i>j</i>}	...	<i>cc</i> _{<i>n</i>}	<i>Z</i>																																																																																								
.	.	.	.	.	.	<i>bb</i> ₁																																																																																								
.	.	.	.	.	.	<i>bb</i> ₂																																																																																								
.	.	.	.	.	.	.																																																																																								
<i>AA</i> ₁	<i>AA</i> ₂	.	<i>AA</i> _{<i>j</i>}	.	<i>AA</i> _{<i>n</i>}	.																																																																																								
.	.	.	.	.	.	<i>bb</i> _{<i>i</i>}																																																																																								
.	.	.	.	.	.	.																																																																																								
.	.	.	.	.	.	<i>bb</i> _{<i>m</i>}																																																																																								

Figure 3.2. Data structure for the K^{th} -best algorithm for Network 1

The K^{th} -best algorithm systematically explores vertices beginning with the basis associated with the optimal solution to the LP created by removing (3.4.3) from the problem (3.4), yielding the SONDP. The following procedure shows the formal K^{th} best algorithm (after Bard, 1998).

Step 0: Set $il=0$, $ir=0$, $count=0$, $LB=-inf$, and $UB=+inf$.

Initialize $Simplex_tableau[150][195]$ from SONDP.

Step 1: Solve SONDP by ILOG/CPLEX Concert Technology; then, assign the objective value to $Array_list[il].SO_TSTT$, store the indices of basic variables in $Array_list[il].BVI[149]$, and set $Array_list[il].picked=0$. Set $il=il+1$. Go to Step 2.

Step 2: If all elements in $Array_list$ are picked, terminate. Otherwise, select the element with the minimal SO_TSTT from $Array_list$; insert this element in $Array_rank[ir]$ and set $ir=ir+1$. If $Array_rank[ir-1].SO_TSTT > LB$, set $LB = Array_rank[ir-1].SO_TSTT$. Go to Step 3.

- Step 3: With the basis from $Array_rank[ir-1].BVI[149]$, determine b_i, x_i^t, y_{ij}^t via Gauss-Jordan elimination on the simplex tableau. Set $Array_rank[ir-1].budget[i]=b_i$ for all i . Go to Step 4.
- Step 4: Solve UONDP with fixed b_i from $Array_rank[ir-1]$ (i.e. solve UO DTA). Set $Array_rank[ir-1].UO_TSTT=UO_TSTT$. If $UB > Array_rank[ir-1].UO_TSTT$, set $UB = Array_rank[ir-1].UO_TSTT$. Go to Step 5.
- Step 5: If $LB \geq UB$, terminate. Otherwise, go to Step 6.
- Step 6: From $Array_rank[ir-1].BVI[149]$ and associated degenerate solutions, generate all adjacent extreme points with non-decreasing objective value. Only adjacent extreme points, which have their objective values $\leq UB$, are inserted into $Array_list[il]$. Go to Step 7.
- Step 7: Set $count = count+1$, and go to Step 2.

From our preliminary computational experience in Section 5.1.2.1 in Chapter 5, the degeneracy leads to extremely long computational time, so we modified the K^{th} -best algorithm by ignoring the existence of degeneracy. This leads to a heuristic with a deterministic error bound. Due to the modification to the original algorithm, the exact solutions cannot be guaranteed. The deterministic error bound of BLPNDP solution is determined from the difference of upper and lower bounds. The algorithm gives the valid upper bound; however, the lower bound from the algorithm is no longer a valid one because of the modification. Thus, we employ a known lower bound, the objective value of SONDP. Recall that SONDP is less constrained than BLPNDP.

In light of the modified K^{th} best algorithm, a possible heuristic for multi-destination problem is the following. The multi-destination SONDP (Li et al., 2003) replaces the single-destination SONDP in Steps 0 and 1, and in Step 4, the simulation-based UO DTA such as VISTA (Ziliaskopoulos and Waller, 2000) replaces the UO DTA LP. Essentially, this iterative heuristic can be considered as a descent method employing the UO DTA to evaluate all candidate solutions. However, a move that transforms a current solution to the new solution requires the generation of all adjacent extreme points

from the simplex tableau of SONDP and the associated matrix manipulation of the large simplex tableau. This is exacerbated by the tedious preprocessing effort in setting up the initial simplex tableau. Also, the solutions obtained from such descent method are normally locally optimal (Glover and Laguna, 1997). Thus, this possible heuristic was not pursued, and we chose to adopt metaheuristics or global search methods that are relatively easier to implement, have less computational effort in obtaining a move from a current solution, are widely used in the literature, and does not get trapped in a local optimum. In Chapter 4, we describe the development of four practical metaheuristics. The next section describes the second analytical solution method for BLPNDP.

3.4.1.2 KKT-Based MIP Reformulation

This approach involves replacing the follower's program of problem (3.32) with its KKT conditions, resulting in a single-level non-linear program (3.35), called the explicit representation of the L-BLPP (Bard, 1998):

$$\min_{x \in X} F(x, y) = c_1x + d_1y \quad (3.35.1)$$

$$\text{subject to} \quad A_1x + B_1y \leq b_1 \quad (3.35.2)$$

$$uB_2 - v = -d_2 \quad (3.35.3)$$

$$u(b_2 - A_2x - B_2y) + vy = 0 \quad (3.35.4)$$

$$A_2x + B_2y \leq b_2 \quad (3.35.5)$$

$$x \geq 0, y \geq 0, u \geq 0, v \geq 0 \quad (3.35.6)$$

where $u \in R^q$ is the vector of dual variables of the follower's program (3.32.3)-(3.32.4), and $v \in R^m$ is the vector of slack variables for the dual feasibility. This mathematical program contains all linear relationships except the nonlinearity in Constraint (3.35.4). Fortuny-Amat and McCarl (1981) reformulated this program as an equivalent MIP as follows. Let $g_i(x, y) \geq 0, i = 1, \dots, q + m$, represent the elements of the column vectors $(b_2 - A_2x - B_2y)$ and y , and let $u'_i, i = 1, \dots, q + m$, represent the elements of the row vectors u and v . Then, the complementary slackness simply means $u'_i g_i(x, y) = 0$,

$i = 1, \dots, q + m$. The nonlinear constraint (3.35.4) is replaced by the following set of inequalities:

$$u'_i \leq MM \times z_i$$

$$g_i \leq MM \times (1 - z_i)$$

where MM is a sufficiently large constant and $z_i \in \{0,1\}, i = 1, \dots, q + m$.

The resulting MIP contains $q+m$ additional variables and $2(q+m)$ additional constraints. This can be solved by a standard zero-one MIP solver (CPLEX in this study). The next section describes the solution method for SLP2-BLPNDP.

3.4.2 Exact Solution Method for SLP2-BLPNDP

The modified K^{th} best algorithm can be applied to SLP2-BLPNDP (3.9), since the multiple nested programs in SLP2-BLPNDP can be represented as a single nested program:

$$\begin{aligned} \min_{x,y} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega} \\ \text{Subject to} \quad CTM^\omega(x, y, b) \quad \forall \omega \in \Omega \end{aligned}$$

However, the modified K^{th} best algorithm is a heuristic and cannot guarantee an exact optimal solution as discussed previously. The KKT-based MIP reformulation is still applicable to (3.9) by replacing the follower's programs (3.9.3)-(3.9.4) with their KKT conditions. This yields a single-level non-linear program, which can further be linearized. The final program is an MIP:

$$\min_b \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega}) \quad (3.36.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.36.2)$$

$$KKT_UODTA^\omega(x, y, \pi, z, b) \quad \forall \omega \in \Omega \quad (3.36.3)$$

$KKT_UODTA^\omega(x, y, \pi, z, b)$ represents three sets of linear constraints: primal feasibility (3.37), dual feasibility (3.38) and linearized complementary slackness (3.40).

Primal feasibility:

$$x_i^{t,\omega} - x_i^{t-1,\omega} + \sum_{(i,j) \in FS(i)} y_{ij}^{t-1,\omega} - \sum_{(j,i) \in RS(i)} y_{ji}^{t-1,\omega} = \tilde{d}_i^{t,\omega} \quad \forall i \in C \setminus C_S, t \in T \quad : \pi_i^{0,t,\omega} \quad (3.37.1)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^{t,\omega} - x_i^{t,\omega} \leq 0 \quad \forall i \in C \setminus C_S, t \in T \quad : \pi_i^{1,t,\omega} \quad (3.37.2)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^{t,\omega} \leq \delta_i^t (N_i^t + \chi_i \cdot b_i - x_i^{t,\omega}) \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad : \pi_i^{2,t,\omega} \quad (3.37.3)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^{t,\omega} \leq Q_i^t + \phi_i \cdot b_i \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad : \pi_i^{3,t,\omega} \quad (3.37.4)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^{t,\omega} \leq Q_i^t + \phi_i \cdot b_i \quad \forall i \in C \setminus C_S, t \in T \quad : \pi_i^{4,t,\omega} \quad (3.37.5)$$

$$x_i^{0,\omega} = \zeta_i \quad \forall i \in C \setminus C_S \quad (3.37.6)$$

$$y_{ij}^{0,\omega} = 0 \quad \forall (i, j) \in E \quad (3.37.7)$$

$$x_i^{|T|,\omega} = 0 \quad \forall i \in C \setminus C_S \quad (3.37.8)$$

$$x_i^{t,\omega} \geq 0 \quad \forall i \in C \setminus C_S, t \in T \quad (3.37.9)$$

$$y_{ij}^{t,\omega} \geq 0 \quad \forall (i, j) \in E, t \in T \quad (3.37.10)$$

Dual feasibility:

$$\pi_i^{0,t,\omega} - \pi_i^{0,t+1,\omega} + \pi_i^{1,t,\omega} \leq 0 \quad \forall i \in C_R, t \in T \setminus \{|T|\} \quad : x_i^{t,\omega} \quad (3.38.1)$$

$$\pi_i^{0,t,\omega} - \pi_i^{0,t+1,\omega} + \pi_i^{1,t,\omega} - \delta_i^t \pi_i^{2,t,\omega} \leq 0 \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \setminus \{|T|\} : x_i^{t,\omega} \quad (3.38.2)$$

$$\pi_i^{0,t+1,\omega} - \pi_j^{0,t+1,\omega} - \pi_i^{1,t,\omega} - \delta_i^t \pi_j^{2,t,\omega} - \pi_j^{3,t,\omega} - \pi_i^{4,t,\omega} \leq 0 \quad \forall (i, j) \in E \setminus E_S, t \in T \setminus \{|T|\} : y_{ij}^{t,\omega} \quad (3.38.3)$$

$$\pi_i^{0,t+1,\omega} - \pi_i^{1,t,\omega} - \pi_i^{4,t,\omega} \leq M_t \quad \forall (i, j) \in E_S, t \in T \setminus \{|T|\} \quad : y_{ij}^{t,\omega} \quad (3.38.4)$$

$$\pi_i^{0,t,\omega} \text{ unrestricted in sign } \forall i \in C \setminus C_S, t \in T \quad (3.38.5)$$

$$\pi_i^{1,t,\omega}, \pi_i^{4,t,\omega} \geq 0 \quad \forall i \in C \setminus C_S, t \in T \quad (3.38.6)$$

$$\pi_i^{2,t,\omega}, \pi_i^{3,t,\omega} \geq 0 \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad (3.38.7)$$

Complementary slackness:

Let $g_i^{1,t,\omega} \geq 0$ denote the inequality (3.37.2) $\forall i \in C \setminus C_S, t \in T$; $g_i^{2,t,\omega} \geq 0$ denote (3.37.3) $\forall i \in C \setminus (C_R \cup C_S), t \in T$; $g_i^{3,t,\omega} \geq 0$ denote (3.37.4) $\forall i \in C \setminus (C_R \cup C_S), t \in T$; and $g_i^{4,t,\omega} \geq 0$ denote (3.37.5) $\forall i \in C \setminus C_S, t \in T$. Let $g_i^{t,\omega} \geq 0$ denote the inequality (3.38.1)-(3.38.2) $\forall i \in C \setminus C_S, t \in T \setminus \{|T|\}$ and $g_{ij}^{t,\omega} \geq 0$ denote (3.38.3)-(3.38.4) $\forall (i, j) \in E, t \in T \setminus \{|T|\}$. The complementary slackness conditions are:

$$g_i^{1,t,\omega} \cdot \pi_i^{1,t,\omega} = 0 \quad \forall i \in C \setminus C_S, t \in T \quad (3.39.1)$$

$$g_i^{2,t,\omega} \cdot \pi_i^{2,t,\omega} = 0 \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad (3.39.2)$$

$$g_i^{3,t,\omega} \cdot \pi_i^{3,t,\omega} = 0 \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad (3.39.3)$$

$$g_i^{4,t,\omega} \cdot \pi_i^{4,t,\omega} = 0 \quad \forall i \in C \setminus C_S, t \in T \quad (3.39.4)$$

$$g_i^{t,\omega} \cdot x_i^{t,\omega} = 0 \quad \forall i \in C \setminus C_S, t \in T \setminus \{|T|\} \quad (3.39.5)$$

$$g_{ij}^{t,\omega} \cdot y_{ij}^{t,\omega} = 0 \quad \forall (i, j) \in E, t \in T \setminus \{|T|\} \quad (3.39.6)$$

The nonlinear conditions (3.39.1)-(3.39.6) can be linearized by defining the following binary variables: $z_i^{1,t,\omega} \quad \forall i \in C \setminus C_S, t \in T$; $z_i^{2,t,\omega} \quad \forall i \in C \setminus (C_R \cup C_S), t \in T$; $z_i^{3,t,\omega} \quad \forall i \in C \setminus (C_R \cup C_S), t \in T$; $z_i^{4,t,\omega} \quad \forall i \in C \setminus C_S, t \in T$; $z_i^{t,\omega} \quad \forall i \in C \setminus C_S, t \in T \setminus \{|T|\}$; and $z_{ij}^{t,\omega} \quad \forall (i, j) \in E, t \in T \setminus \{|T|\}$. With sufficiently large constant MM , the linearized complementary slackness conditions are:

$$\pi_i^{1,t,\omega} \leq MM \cdot z_i^{1,t,\omega} \quad \forall i \in C \setminus C_S, t \in T \quad (3.40.1)$$

$$g_i^{1,t,\omega} \leq MM \cdot (1 - z_i^{1,t,\omega}) \quad \forall i \in C \setminus C_S, t \in T \quad (3.40.2)$$

$$\pi_i^{2,t,\omega} \leq MM \cdot z_i^{2,t,\omega} \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad (3.40.3)$$

$$g_i^{2,t,\omega} \leq MM \cdot (1 - z_i^{2,t,\omega}) \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad (3.40.4)$$

$$\pi_i^{3,t,\omega} \leq MM \cdot z_i^{3,t,\omega} \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad (3.40.5)$$

$$g_i^{3,t,\omega} \leq MM \cdot (1 - z_i^{3,t,\omega}) \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad (3.40.6)$$

$$\pi_i^{4,t,\omega} \leq MM \cdot z_i^{4,t,\omega} \quad \forall i \in C \setminus C_S, t \in T \quad (3.40.7)$$

$$g_i^{4,t,\omega} \leq MM \cdot (1 - z_i^{4,t,\omega}) \quad \forall i \in C \setminus C_S, t \in T \quad (3.40.8)$$

$$x_i^{t,\omega} \leq MM \cdot z_i^{t,\omega} \quad \forall i \in C \setminus C_S, t \in T \setminus \{|T|\} \quad (3.40.9)$$

$$g_i^{t,\omega} \leq MM \cdot (1 - z_i^{t,\omega}) \quad \forall i \in C \setminus C_S, t \in T \setminus \{|T|\} \quad (3.40.10)$$

$$y_{ij}^{t,\omega} \leq MM \cdot z_{ij}^{t,\omega} \quad \forall (i, j) \in E, t \in T \setminus \{|T|\} \quad (3.40.11)$$

$$g_{ij}^{t,\omega} \leq MM \cdot (1 - z_{ij}^{t,\omega}) \quad \forall (i, j) \in E, t \in T \setminus \{|T|\} \quad (3.40.12)$$

The next section describes the solution method for RO-BLPNDP.

3.4.3 Exact Solution Method for RO-BLPNDP

The continuous quadratic-linear bi-level program of RO-BLPNDP (3.10) contains all linear terms except the quadratic term in the upper-level objective function. It can be shown that the upper-level objective function is convex quadratic as follows. Since the weight λ , the probability $p^\omega \forall \omega \in \Omega$ and the squared term $(D^\omega)^2 \forall \omega \in \Omega$ are non-negative, these can be re-arranged as $y^T Q y$ where y is the vector of variables and Q is the vector of coefficients. Clearly, the quantity $y^T Q y$ is non-negative ($y^T Q y \geq 0$) for all non-zero real vector y ; thus, Q is positive semidefinite and the quadratic upper-level objective function is convex.

The modified K^{th} best algorithm cannot be applied to RO-BLPNDP because the optimal solution is not always at the extreme point of the underlying constraint region. However, the KKT-based MIP reformulation is still applicable. The nested linear programs (3.10.5)-(3.10.6) are replaced with their linearized KKT conditions ((3.37), (3.38) and (3.40)), yielding the convex mixed-integer nonlinear program (MINLP):

$$\min_{b, D^\omega} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega}) + \lambda \sum_{\omega \in \Omega} (p^\omega \cdot (D^\omega)^2) \quad (3.41.1)$$

$$\text{subject to} \quad BUDGET_CONSTRAINT(b) \quad (3.41.2)$$

$$D^\omega \geq \sum_{(i,j) \in E_S} \sum_{t \in T} (t \cdot y_{ij}^{t,\omega}) - TSTT^{TARGET} \quad \forall \omega \in \Omega \quad (3.41.3)$$

$$D^\omega \geq 0 \quad \forall \omega \in \Omega \quad (3.41.4)$$

$$KKT_UODTA^{\omega}(x, y, \pi, z, b) \quad \forall \omega \in \Omega \quad (3.41.5)$$

The only non-linearity in (3.41) is the upper-level objective function, which is convex quadratic. Thus, the global optimum can be found by any MINLP solver (GAMS/DICOPT in this study). The next section describes the exact solution methods for the SSD and NDP-SSD models.

3.5 Exact Solution Methods for SSD and NDP-SSD Models

It is well known that mixed integer programming problems are NP-hard. This is exacerbated by formulating a MIP as a bi-level programming problem, which is strongly NP-hard (Bard, 1998). Hence, there is not an efficient algorithm to solve the SSD and NDP-SSD models. Since the main goals of this analytical study are the development of new formulations, the comparison among the NDP, SSD and NDP-SSD models and the comparison between sequential and integrated approaches, we do not emphasize on efficient algorithms and only employ certain exponential exact solution methods.

The single-level models (SOSSD, UOSSD, SLP2-SOSSD, SLP2-UOSSD, SONDP-SSD, UONDP-SSD, SLP2-SONDP-SSD and SLP2-UONDP-SSD) are mixed-integer programs, and can be directly solved by any MIP solver (CPLEX in this study). The solution methods for mixed-integer bi-level programming models (BLPSSD, SLP2-BLPSSD, RO-BLPSSD, BLPNDP-SSD, SLP2-BLPNDP-SSD, RO-BLPNDP-SSD) are described, respectively.

3.5.1 Exact Solution Method for BLPSSD and BLPNDP-SSD

In BLPSSD (3.15) and BLPNDP-SSD (3.25), the upper-level decision variables are both binary and continuous (i.e. mixed zero-one), and the lower-level variables are continuous. All terms in (3.15) and (3.25) are linear. As such, the problems (3.15) and (3.25) belong to the class of mixed-zero-one continuous linear bi-level programming problem (MZCL-BLPP), whose standard form can be stated as (after Bard, 1998):

$$\min_x F(x, y) = c_{11}x_1 + c_{12}x_2 + d_1y \quad (3.42.1)$$

$$\text{subject to} \quad A_{11}x_1 + A_{12}x_2 + B_1y \leq b_1 \quad (3.42.2)$$

$$x_1 \geq 0, x_2 \in \{0,1\} \quad (3.42.3)$$

$$\min_y f(y) = d_2y \quad (3.42.4)$$

$$\text{subject to} \quad A_{21}x_1 + A_{22}x_2 + B_2y \leq b_2 \quad (3.42.5)$$

$$y \geq 0 \quad (3.42.6)$$

where x_1 is an n_1 -dimensional vector of continuous variables; x_2 is an n_2 -dimensional vector of binary variables; $x = (x_1, x_2)$ is an n -dimensional vector of upper-level decision variables; $n = n_1 + n_2$; $y = m$ -dimensional vector of continuous variables; and all vectors and matrices are of conformal dimension.

A naive approach to solve this MZCL-BLPP is to employ a branch-and-bound algorithm where an L-BLPP is a subproblem at a node of the branch-and-bound tree. This is extremely difficult, since an L-BLPP is already strongly NP-hard. Further, the typical fathoming rules employed in mixed-integer programming are not always applicable to the mixed-integer linear BLPP (Bard, 1998). Instead, we will show that MZCL-BLPP can be reduced to the following equivalent parametric L-BLPP (L-BLPP(M)) for a sufficiently large constant M :

$$\min_x F(x, y) = c_{11}x_1 + c_{12}x_2 + d_1y + Me_{n_2}^T u \quad (3.43.1)$$

$$\text{subject to} \quad A_{11}x_1 + A_{12}x_2 + B_1y \leq b_1 \quad (3.43.2)$$

$$x_1 \geq 0, \quad 0 \leq x_2 \leq e_{n_2} \quad (3.43.3)$$

$$\min_{y,u} f(y) = d_2y - e_{n_2}^T u \quad (3.43.4)$$

$$\text{subject to} \quad A_{21}x_1 + A_{22}x_2 + B_2y \leq b_2 \quad (3.43.5)$$

$$u \leq x_2 \quad (3.43.6)$$

$$u \leq e_{n_2} - x_2 \quad (3.43.7)$$

$$y \geq 0 \quad (3.43.8)$$

where e_{n_2} is the vector of ones with dimension n_2 ; and u is an n_2 -dimensional vector of continuous variables. Appendix A shows a proof that the MZCL-BLPP can be reduced to an equivalent L-BLPP(M). The proof is directly based on the work by Bard (1998) and Kalantari and Rosen (1982), who show the reduction of ZC-L-BLPP to an equivalent L-BLPP(M) as discussed in Section 2.8. To solve such L-BLPP(M), the MIP reformulation technique used in Section 3.4.1.2 can be adopted here. The nested linear program (3.43.4)-(3.43.8) is replaced with its linearized KKT conditions, yielding a single-level MIP, which can be solved by CPLEX. Since the value of M in L-BLPP(M) is not known, Appendix A describes a penalty-based algorithm that solves a sequence of L-BLPP(M) until the upper-level zero-one variables (i.e. variables s_i) are integral.

3.5.2 Exact Solution Method for SLP2-BLPSSD and SLP2-BLPNDP-SSD

The SLP2-BLPSSD (3.20) and SLP2-BLPNDP-SSD (3.30) contain multiple nested programs; however, these nested programs are equivalent to a single nested program. For example, the multiple nested programs in SLP2-BLPSSD are equivalent to:

$$\begin{aligned} \min_{x,y} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega} \\ \text{subject to} \quad CTM^\omega(x, y, s, p) \quad \forall \omega \in \Omega \end{aligned}$$

Thus, SLP2-BLPSSD and SLP2-BLPNDP-SSD are MZCL-BLPPs, which can be reduced to equivalent parametric L-BLPPs. For example, the SLP2-BLPSSD (3.20) can be reduced to the L-BLPP(M) (3.44) for sufficiently large M .

$$\min_{s,p} \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot t \cdot y_{ij}^{t,\omega}) + \sum_{\omega \in \Omega} \sum_{i \in INTERSECTION} \sum_{j \in STP_i} M \cdot u_j^\omega \quad (3.44.1)$$

$$\text{subject to} \quad \sum_{i \in STP_j} s_i = 1 \quad \forall j \in INTERSECTION \quad (3.44.2)$$

$$p_{ij}^t = \sum_{k \in STP_j: \text{phase}(k,t) \in P(i,j)} s_k \quad \forall (i,j) \in RS(j), \forall j \in INTERSECTION, \forall t \in T \quad (3.44.3)$$

$$0 \leq s_i \leq 1 \quad \forall i \in \cup_{j \in INTERSECTION} STP_j \quad (3.44.4)$$

$$p_{ij}^t \geq 0 \quad \forall (i,j) \in RS(j), \forall j \in INTERSECTION, \forall t \in T \quad (3.44.5)$$

$$\min_{x,y,u} \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t,\omega} - \sum_{i \in INTERSECTION} \sum_{j \in STP_i} u_j^\omega \quad \forall \omega \in \Omega \quad (3.44.6)$$

$$\text{subject to} \quad CTM^\omega(x, y, s, p) \quad \forall \omega \in \Omega \quad (3.44.7)$$

$$u_i^\omega \leq s_i \quad \forall i \in \cup_{j \in INTERSECTION} STP_j, \quad \forall \omega \in \Omega \quad (3.44.8)$$

$$u_i^\omega \leq 1 - s_i \quad \forall i \in \cup_{j \in INTERSECTION} STP_j, \quad \forall \omega \in \Omega \quad (3.44.9)$$

To solve such L-BLPP(M) (3.44), the nested linear programs (3.44.6)-(3.44.9) are replaced with their linearized KKT conditions, yielding a single-level mixed-integer program. This transformation is similar to Section 3.4.2. The next section describes the solution method for RO-BLPSSD and RO-BLPNDP-SSD.

3.5.3 Exact Solution Method for RO-BLPSSD and RO-BLPNDP-SSD

The RO-BLPSSD (3.21) and RO-BLPNDP-SSD (3.31) have linear terms except the quadratic upper-level objective functions, which are convex as discussed in Section 3.4.3. The upper-level decision variables are mixed zero-one variables, and the lower-level variables are continuous. Then, the problems (3.21) and (3.31) belong to the class of mixed-zero-one continuous quadratic-linear bi-level programming problem (MZO-QL-BLPP), whose standard form can be stated as follows (after Bard, 1998):

$$\min_x F(x, y) = c_{11}x_1 + c_{12}x_2 + d_1y + \frac{1}{2}y^T Qy \quad (3.45.1)$$

$$\text{subject to} \quad A_{11}x_1 + A_{12}x_2 + B_1y \leq b_1 \quad (3.45.2)$$

$$x_1 \geq 0, x_2 \in \{0,1\} \quad (3.45.3)$$

$$\min_y f(y) = d_2y \quad (3.45.4)$$

$$\text{subject to} \quad A_{21}x_1 + A_{22}x_2 + B_2y \leq b_2 \quad (3.45.5)$$

$$y \geq 0 \quad (3.45.6)$$

where x_1 is an n_1 -dimensional vector of continuous variables; x_2 be an n_2 -dimensional vector of binary variables; $x = (x_1, x_2)$; $n = n_1 + n_2$; y is an m -dimensional vector of continuous variables; and other vectors and matrices are of conformal dimension.

We will show that the problem MZC-QL-BLPP (3.45) can be reduced to the following equivalent parametric quadratic-linear bi-level programming problem, QL-BLPP(M), for a sufficiently large constant M :

$$\min_x F(x, y) = c_{11}x_1 + c_{12}x_2 + d_1y + \frac{1}{2}y^T Qy + Me_{n_2}^T u \quad (3.46.1)$$

$$\text{subject to} \quad A_{11}x_1 + A_{12}x_2 + B_1y \leq b_1 \quad (3.46.2)$$

$$x_1 \geq 0, \quad 0 \leq x_2 \leq e_{n_2} \quad (3.46.3)$$

$$\min_{y, u} f(y) = d_2y - e_{n_2}^T u \quad (3.46.4)$$

$$\text{subject to} \quad A_{21}x_1 + A_{22}x_2 + B_2y \leq b_2 \quad (3.46.5)$$

$$u \leq x_2 \quad (3.46.6)$$

$$u \leq e_{n_2} - x_2 \quad (3.46.7)$$

$$y \geq 0 \quad (3.46.8)$$

Appendix B shows a proof that MZC-QL-BLPP can be reduced to an equivalent QL-BLPP(M). The proof is directly based on the work by Bard (1998) and Kalantari and Rosen (1982) as discussed in Section 2.8. QL-BLPP(M) can be solved by first replacing the nested linear program (3.46.4)-(3.46.8) with its linearized KKT conditions, yielding a single-level mixed integer non-linear program (MINLP) with all linear terms except the convex quadratic objective function. Thus, we can determine a global optimal solution of such MINLP by GAMS/DICOPT. Since the value of M in QL-BLPP(M) is not known, Appendix B describes a penalty-based algorithm that solves a sequence of QL-BLPP(M) until the upper-level zero-one variables (i.e. variables s_i) are integral.

3.6 Summary

In this chapter, several mathematical models are developed for network design problem (NDP), DTA-based signal setting design problem (SSD) and the combined NDP, SSD and DTA problem (NDP-SSD). For NDP, we propose three new optimization models (BLPNDP, SLP2-BLPNDP and RO-BLPNDP) based on DTA LP models when

accounting for the inherent bi-level nature of the problem. SLP2-BLPNDP is a two-stage stochastic program extended from BLPNDP when accounting for long-term O-D demand uncertainty. RO-BLPNDP is built upon SLP2-BLPNDP by incorporating the risk term into the objective function. Given a degree of robustness, RO-BLPNDP provides the most robust NDP solution that is the least sensitive to the variation of O-D demand. For SSD and NDP-SSD, the new models are extended from the existing deterministic system-optimal SSD and NDP-SSD models. Like NDP, we formulate the single-level SSD and NDP-SSD models (UOSSD, SLP2-SOSSD, SLP2-UOSSD, UONDP-SSD, SLP2-SONDP-SSD, and SLP2-UONDP-SSD) and the bi-level models (BLPSSD, SLP2-BLPSSD, RO-BLPSSD, BLPNDP-SSD, SLP2-BLPNDP-SSD and RO-BLPNDP-SSD). We also show that the robust NDP, SSD and NDP-SSD models must be bi-level in order to be meaningful. In SSD, cycle lengths, green splits, time offsets and phase sequencing are optimized with fixed signal phasing; thus, the models capture the effect of signal coordination. We allow intersections to have different cycle lengths, since the commonly used assumption of common cycle length is considered too constrained. Subsequently, the exact solution methods for NDP (the K^{th} best algorithm and the KKT-based MIP reformulation) for all models are described. For the exact solution methods for SSD and NDP-SSD, we show that a mixed-zero-one continuous linear bi-level program can be reduced to an equivalent parametric linear bi-level program, and a mixed-zero-one continuous quadratic-linear bi-level program can be reduced to an equivalent parametric quadratic-linear bi-level program.

The work in this chapter provides new perspectives on traditionally difficult and intractable problems. Although the models are limited to single-destination, small networks and may not be practical, they allow us to gain insights into the problems by numerically comparing the deterministic, stochastic and robust formulations as well as evaluating a sequential approach against the combined approach in Chapter 5. The work in this chapter also helps lay down the basis for developing more practical metaheuristic methods in the next chapter.

Chapter 4. Metaheuristic Approach

In the previous chapter, the bi-level formulations and exact solution methods are proposed for the network design problem (NDP), the DTA-based signal setting design problem (SSD) and the combined NDP, SSD and DTA problem (NDP-SSD) under deterministic, stochastic and robust optimization schemes. These can provide a great deal of insights into the problems; however, the limitations due to the underlying DTA LP models prohibit the useful, practical application on multi-destination larger-size networks. In this chapter, different practical metaheuristic methods that can produce solutions beyond local optimality are developed. Decision variables of the NDP, SSD and NDP-SSD are first discussed, followed by the solution representations and decoding procedures. Then, the objective functional evaluation procedures for the deterministic, stochastic and robust optimization schemes are described. Subsequently, five metaheuristic algorithms are given: simulated annealing (SA), modified simulated annealing (MSA), genetic algorithm (GA), reactive tabu search (RTS) and random search (RS).

4.1 Decision Variables

The decision variables of NDP, SSD and NDP-SSD are described, respectively.

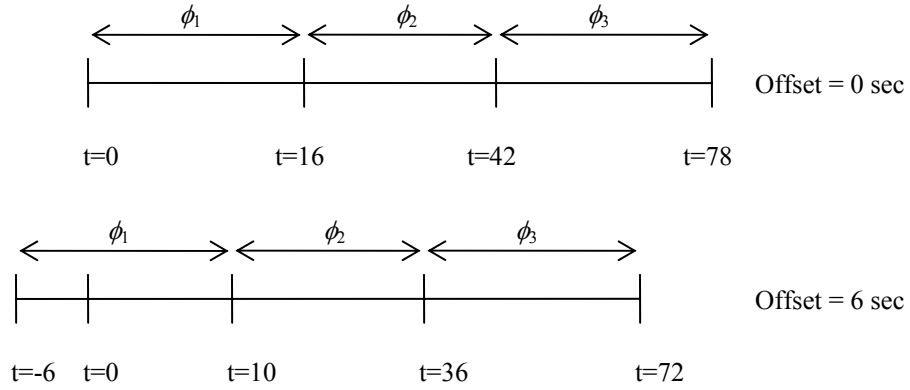
4.1.1 Decision Variables of NDP

From the analytical results on Network 1 in Chapter 5, the available budgets are always exhausted for the optimal solutions; thus, the decision variables of NDP can be seen as fractional values of the total available budget (i.e., b_c in Chapter 3 can be represented as $fb_c \cdot TAB$, where fb_c is the fraction of TAB assigned to cell c). Since all cells within a link should be assigned an equal amount of budget to avoid a bottleneck, we consider a variable associated with a link rather than a cell. Formally, we define the variables fbp_i as the fraction of TAB assigned to each cell of link i , the variables fbp'_i as

the fraction of TAB assigned to link i . As such, $fbp_i = fbp'_i / N_fbp_i$, where N_fbp_i is the number of cells on link i . The next section describes the decision variables of SSD.

4.1.2 Decision Variables of SSD

The decision variables are cycle lengths, green splits, phase sequences and time offsets of pre-specified intersections in the network. We assume that signal phases as well as possible phase sequencing are exogenously determined. Additionally, the meaning of time offset is illustrated in Figure 4.1.



Note: Three signal phases are assumed for 3-leg intersection.
 $t=0$ is the beginning of the simulation period.
 ϕ_i is the signal phase time i .

Figure 4.1. Example of Time Offset of a Three-Leg Intersection

Based on the work by Park (1998), we define the vector of fractional variables f , whose elements are bounded between 0 and 1:

$$f_{3-leg} = \{f_1, f_2, f_3, f_4, f_5\} \text{ for a three-leg intersection} \quad (4.1.1)$$

$$f_{4-leg} = \{f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8\} \text{ for a four-leg intersection} \quad (4.1.2)$$

$$f_{5-leg} = \{f_1, f_2, f_3, f_4, f_5, f_6, f_7\} \text{ for a five-leg intersection} \quad (4.1.3)$$

As such, the size of the vector of fractional variables for SSD ($fssd$) depends on the number and types of intersections in the network. For example, the vector $fssd$ for a network with one of each intersection type is $fssd = \{f_{3-leg}, f_{4-leg}, f_{5-leg}\}$. Next, the

fractional variables of three intersection types and corresponding decoding procedures are described.

4.1.2.1 Three-Leg Intersection

We employ a three-leg intersection with three movement phases in Figure 4.2. The defined vector of fractional variables, f_{3-leg} , are associated with the decision variables in Figure 4.3. The possible phase sequences of three-leg intersection are enumerated in Figure 4.4. The decoding procedure in Figure 4.5 is employed to convert the vector f_{3-leg} to decision variables.

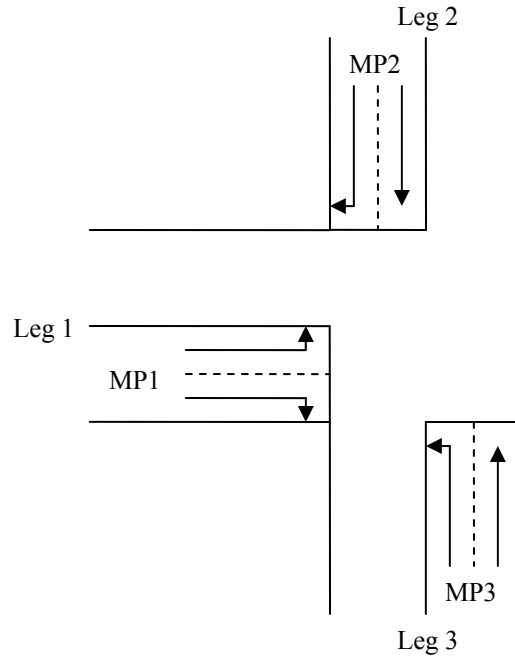
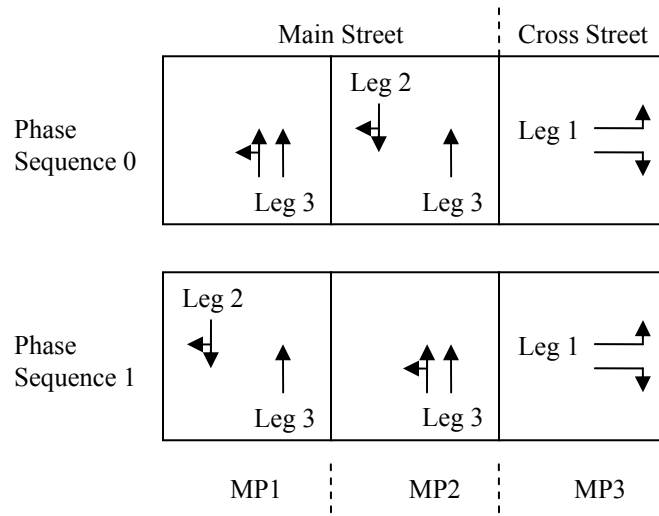


Figure 4.2. Three Movement Phases of Three-Leg Intersection

Fractional Variable	Decision Variables
f_1	Cycle length
f_2	Main (legs 2-3) and cross (leg 1) street phase times
f_3	Durations of movement phases 1 and 2
f_4	Signal phase sequence
f_5	Time offset

Figure 4.3. Fractional Variables and Decision Variables of Three-Leg Intersection

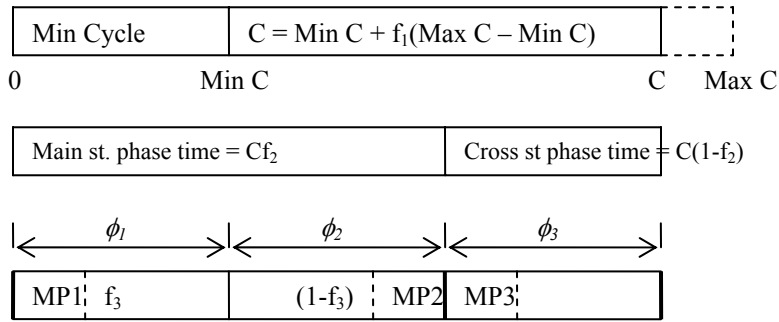


Note: MP = Movement Phase

Figure 4.4. Two Signal Phase Sequences of Three-Leg Intersection

<i>Parameters</i>	
MIN_C	= minimum cycle length in second ($\text{MIN_C} \geq \text{MP}+3\text{Y}+3\text{AR}$)
MAX_C	= maximum cycle length in second
MP1, MP2, MP3	= minimum green times for movement phases 1, 2, and 3, respectively
MP	= minimum phase time = $\text{MP1}+\text{MP2}+\text{MP3}$
Y	= yellow time for each movement phase in second
AR	= all-red time for each movement phase in second
<i>Decision Variables</i>	
Cycle	= $\text{MIN_C} + f_1 \times (\text{MAX_C} - \text{MIN_C})$
Green_Main	= $\text{MP1} + \text{MP2} + f_2 \times (\text{Cycle} - \text{MP} - 3\text{Y} - 3\text{AR})$
Green_Cross	= $\text{Cycle} - 3\text{Y} - 3\text{AR} - \text{Green_Main}$
Green(ϕ_1)	= $\text{MP1} + f_2 \times f_3 \times (\text{Cycle} - \text{MP} - 3\text{Y} - 3\text{AR})$
Green(ϕ_2)	= $\text{Green_Main} - \text{Green}(\phi_1)$
Green(ϕ_3)	= Green_Cross
Phase_Sequence	= $\lfloor f_4 \times 1.9999999 \rfloor$, where $\lfloor a \rfloor$ is the greatest integer $\leq a$.
Offset	= $\lfloor f_5 \times \text{Cycle} \rfloor$

a) Decoding Algorithm for Three-Leg Intersection



b) Determination of Cycle Length and Green Splits

Figure 4.5. Decoding Procedure for Three-Leg Intersection

4.1.2.2 Four-Leg Intersection

A four-leg intersection with eight NEMA movement phases in Figure 4.6 is employed. The defined vector of fractional variables, f_{4-leg} , are associated with the decision variables in Figure 4.7. The possible signal phase sequences of four-leg intersection are enumerated in Figure 4.8. The decoding procedure in Figure 4.9 is used to convert the vector f_{4-leg} to decision variables. As shown in Figure 4.9b, it is possible to account for overlapping movement phases in this scheme; however, for simplicity, we

assume no overlapping movement phases in this study. Specifically, we assume $f_3 = f_4$ and $f_5 = f_6$ in f_{4-leg} .

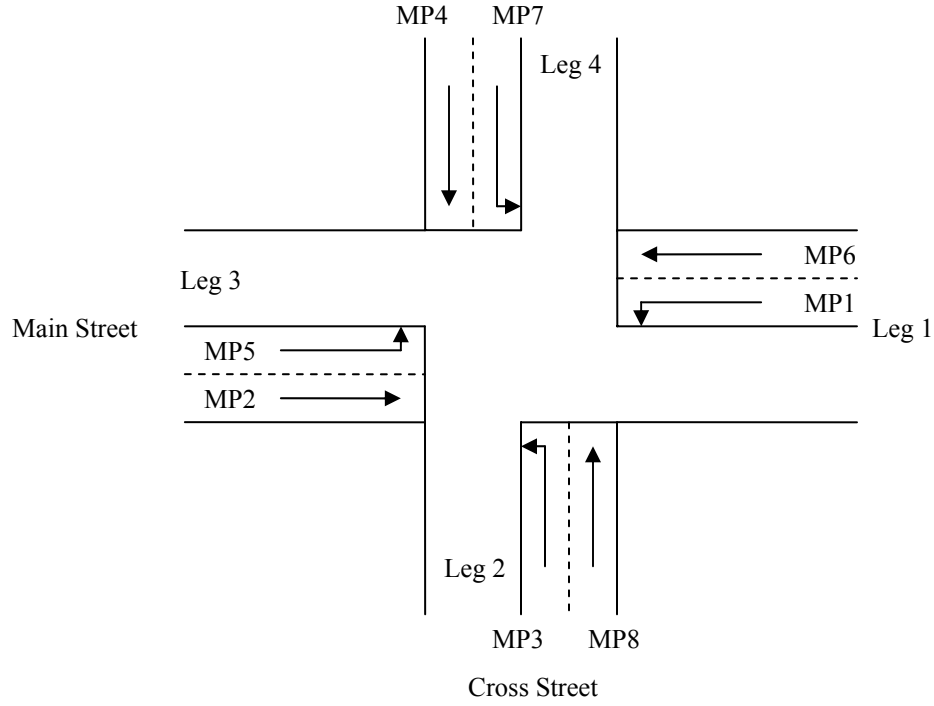


Figure 4.6. Eight NEMA Movement Phases of Four-Leg Intersection

Fractional Variable	Decision Variables
f_1	Cycle length
f_2	Main and cross street phase times
f_3	Duration of movement phases 1 and 2
f_4	Duration of movement phases 5 and 6
f_5	Duration of movement phases 3 and 4
f_6	Duration of movement phases 7 and 8
f_7	Signal phase sequence
f_8	Time offset

Figure 4.7. Fractional Variables and Decision Variables of Four-Leg Intersection

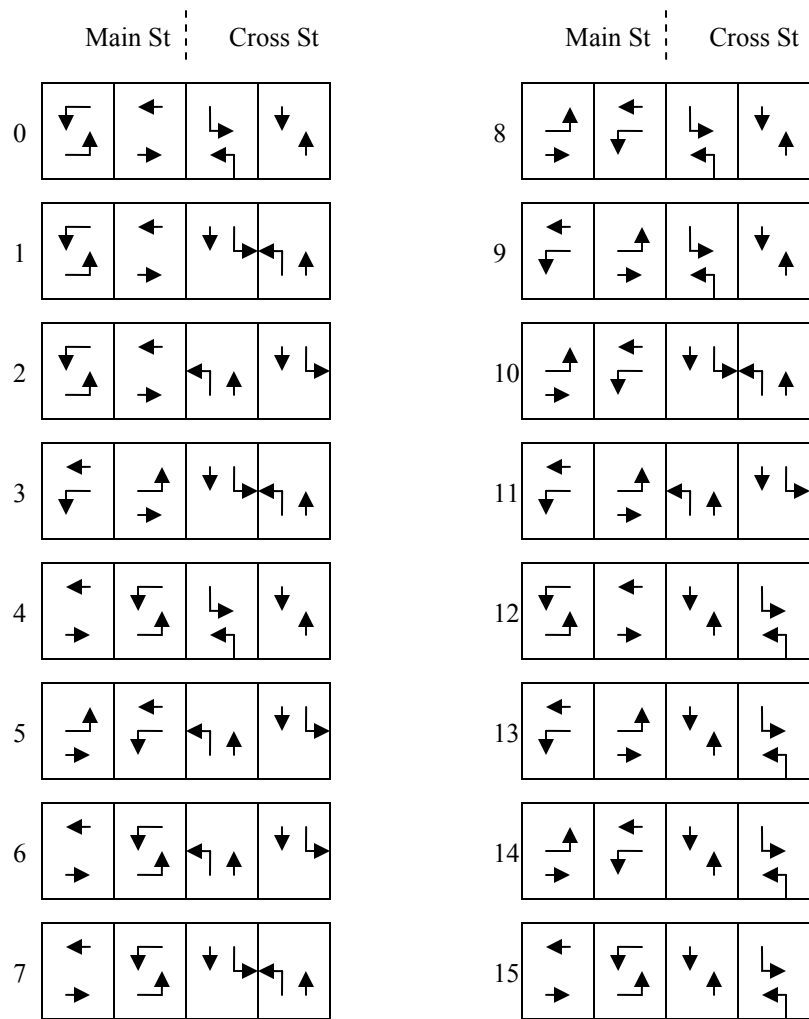
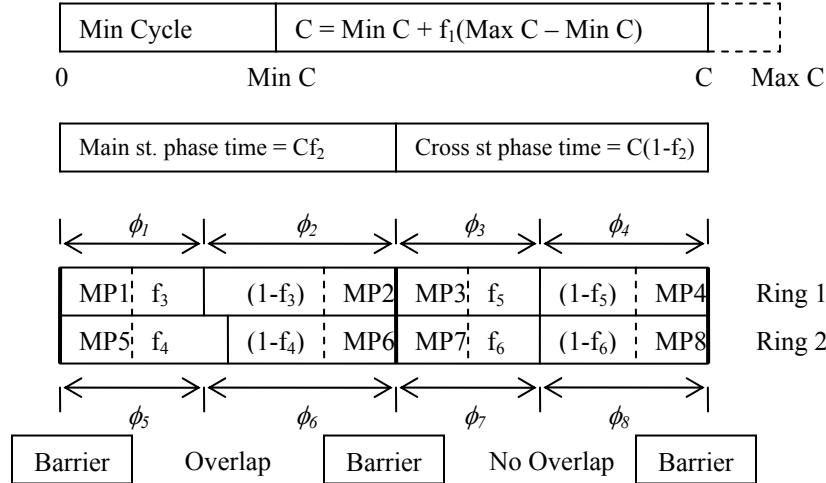


Figure 4.8. Sixteen Signal Phase Sequences of Four-Leg Intersection (Park, 1998)

<i>Parameters</i>	
MIN_C	= minimum cycle length in second ($\text{MIN_C} \geq \text{MP} + 4\text{Y} + 4\text{AR}$)
MAX_C	= maximum cycle length in second
MP1, MP2, ..., MP8	= minimum green times for movement phases 1, 2, ..., 8, respectively
MP	= minimum green phase time = $\max(\text{MP1} + \text{MP2} + \text{MP3} + \text{MP4}, \text{MP5} + \text{MP6} + \text{MP7} + \text{MP8})$
Y	= yellow time for each movement phase in second
AR	= all-red time for each movement phase in second
<i>Decision Variables</i>	
Cycle	= $\text{MIN_C} + f_1 \times (\text{MAX_C} - \text{MIN_C})$
Green_Main	= $\text{MP1} + \text{MP2} + f_2 \times (\text{Cycle} - \text{MP} - 4\text{Y} - 4\text{AR})$
Green_Cross	= $\text{Cycle} - 4\text{Y} - 4\text{AR} - \text{Green_Main}$
Green(ϕ_1) = Green(ϕ_5)	= $\text{MP1} + f_2 \times f_3 \times (\text{Cycle} - \text{MP} - 4\text{Y} - 4\text{AR})$
Green(ϕ_2) = Green(ϕ_6) = Green_Main - Green(ϕ_1)	
Green(ϕ_3) = Green(ϕ_7)	= $\text{MP3} + (1 - f_2) \times f_3 \times (\text{Cycle} - \text{MP} - 4\text{Y} - 4\text{AR})$
Green(ϕ_4) = Green(ϕ_8) = Green_Cross - Green(ϕ_3)	
Phase_Sequence	= $\lfloor f_7 \times 15.9999999 \rfloor$, where $\lfloor a \rfloor$ is the greatest integer $\leq a$.
Offset	= $\lfloor f_8 \times \text{Cycle} \rfloor$

a) Decoding Algorithm for Four-Leg Intersection



Note: In this study, we assume no overlap phases; i.e. $f_3 = f_4$ and $f_5 = f_6$.

b) Determination of Cycle Length and Green Splits

Figure 4.9. Decoding Procedure for Four-Leg Intersection (Park, 1998)

4.1.2.3 Five-leg Intersection

A five-leg intersection with five movement phases in Figure 4.10 is employed. The defined vector of fractional variables, f_{5-leg} , are associated with the decision variables in Figure 4.11. The possible signal phase sequences of five-leg intersection are enumerated in Figure 4.12. The decoding procedure in Figure 4.13 is employed to convert the vector f_{5-leg} to decision variables.

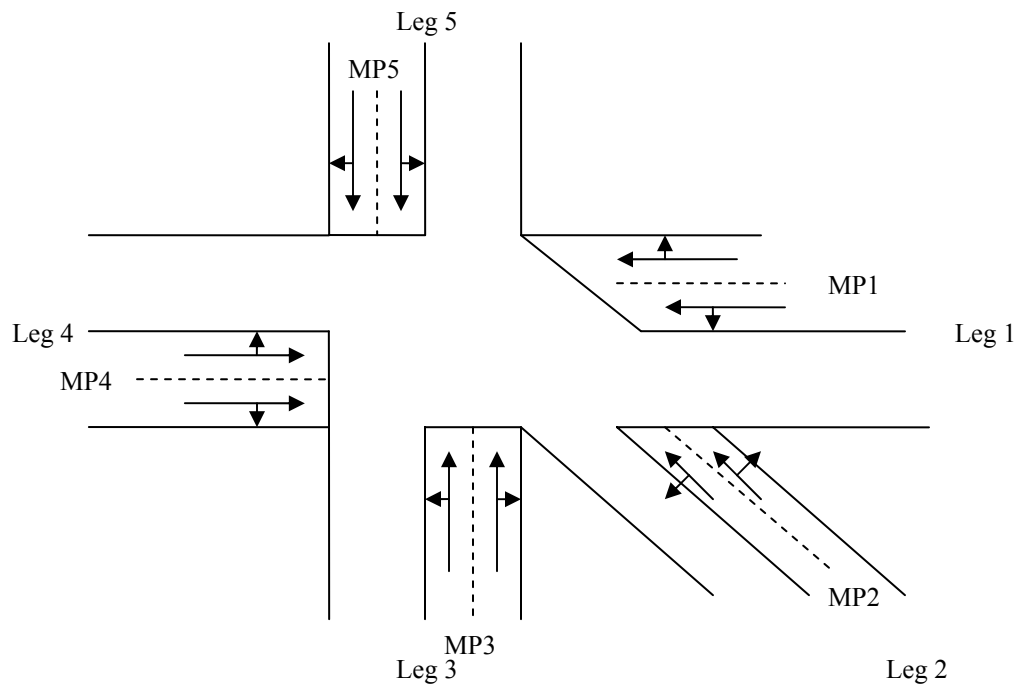


Figure 4.10. Five Movement Phases of Five-Leg Intersection

Fractional Variable	Decision Variables
f_1	Cycle length
f_2	Phases splits for movement phases 1,4 and movement phases 2,3,5
f_3	Phase splits for movement phases 2,3 and movement phase 5
f_4	Duration of movement phases 1 and 4
f_5	Duration of movement phases 2 and 3
f_6	Signal Phase sequence
f_7	Time offset

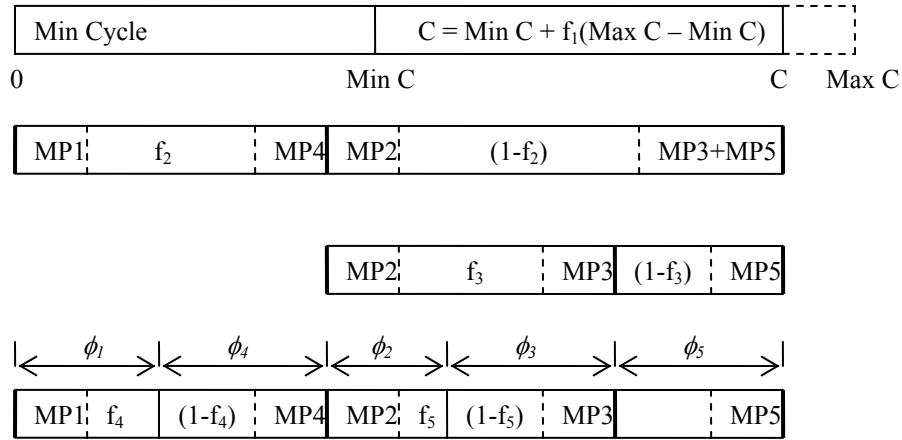
Figure 4.11. Fractional Variables and Decision Variables of Five-Leg Intersection

Phase Sequence	Main Street		Cross Streets		
0	From Leg 1	From Leg 4	From Leg 2	From Leg 3	From Leg 5
1	From Leg 1	From Leg 4	From Leg 2	From Leg 5	From Leg 3
2	From Leg 1	From Leg 4	From Leg 3	From Leg 2	From Leg 5
3	From Leg 1	From Leg 4	From Leg 3	From Leg 5	From Leg 2
4	From Leg 1	From Leg 4	From Leg 5	From Leg 2	From Leg 3
5	From Leg 1	From Leg 4	From Leg 5	From Leg 3	From Leg 2
6	From Leg 4	From Leg 1	From Leg 2	From Leg 3	From Leg 5
7	From Leg 4	From Leg 1	From Leg 2	From Leg 5	From Leg 3
8	From Leg 4	From Leg 1	From Leg 3	From Leg 2	From Leg 5
9	From Leg 4	From Leg 1	From Leg 3	From Leg 5	From Leg 2
10	From Leg 4	From Leg 1	From Leg 5	From Leg 2	From Leg 3
11	From Leg 4	From Leg 1	From Leg 5	From Leg 3	From Leg 2

Figure 4.12. Twelve Signal Phase Sequences of Five-Leg Intersection

<i>Parameters</i> MIN_C = minimum cycle length in second ($\text{MIN_C} \geq \text{MP}+5\text{Y}+5\text{AR}$) MAX_C = maximum cycle length in second MP1, MP2, ..., MP5 = minimum green times for movement phases 1, 2,..., 5, respectively MP = minimum green phase time $= \max(\text{MP1}+\text{MP2}+\text{MP3}+\text{MP4}+\text{MP5})$ Y = yellow time for each movement phase in second AR = all-red time for each movement phase in second
<i>Decision Variables</i> $\text{Cycle} = \text{MIN_C} + f_1 \times (\text{MAX_C} - \text{MIN_C})$ $\text{Green_Leg1,4} = \text{MP1} + \text{MP4} + f_2 \times (\text{Cycle} - \text{MP} - 5\text{Y} - 5\text{AR})$ $\text{Green}(\phi_1) = \text{MP1} + f_2 \times f_4 \times (\text{Cycle} - \text{MP} - 5\text{Y} - 5\text{AR})$ $\text{Green}(\phi_4) = \text{Green_Leg1,4} - \text{Green}(\phi_1)$ $\text{Green_Leg2,3,5} = \text{MP2} + \text{MP3} + \text{MP5} + (1 - f_2) \times (\text{Cycle} - \text{MP} - 5\text{Y} - 5\text{AR})$ $\text{Green}(\phi_5) = \text{MP5} + (1 - f_2) \times (1 - f_3) \times (\text{Cycle} - \text{MP} - 5\text{Y} - 5\text{AR})$ $\text{Green_Leg2,3} = \text{MP2} + \text{MP3} + (1 - f_2) \times f_3 \times (\text{Cycle} - \text{MP} - 5\text{Y} - 5\text{AR})$ $\text{Green}(\phi_2) = \text{MP2} + (1 - f_2) \times f_3 \times f_5 \times (\text{Cycle} - \text{MP} - 5\text{Y} - 5\text{AR})$ $\text{Green}(\phi_3) = \text{Green_Leg2,3} - \text{Green}(\phi_2)$ $\text{Phase_Sequence} = \lfloor f_6 \times 11.9999999 \rfloor$, where $\lfloor a \rfloor$ is the greatest integer $\leq a$. $\text{Offset} = \lfloor f_7 \times \text{Cycle} \rfloor$

a) Decoding Algorithm for Five-Leg Intersection



b) Determination of Cycle Length and Green Splits

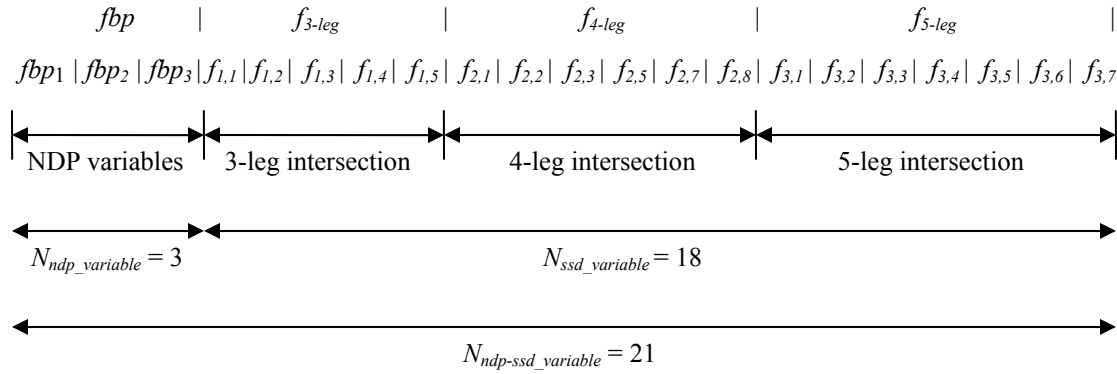
Figure 4.13. Decoding Procedure for Five-Leg Intersection

4.1.3 Decision Variables of NDP-SSD

The decision variables for NDP-SSD are simply the combination of the NDP variables and the SSD variables. To illustrate, let's consider a network with one of each intersection type and 3 pre-specified links for capacity expansion. The vector of fractional variables ($fndp_ssd$) is

$$fndp_ssd = \{fbp, fssd\} = \{fbp, f_{3-leg}, f_{4-leg}, f_{5-leg}\}$$

as elaborated in Figure 4.14. In Figure 4.14, denote by $N_{ndp_variable}$ the number of fractional variables for NDP; $N_{ssd_variable}$ as the number of fractional variables for SSD; and $N_{ndp-ssd_variable}$ as the number of fractional variables for NDP-SSD.



Note: For 4-leg intersection, we assume non-overlapping movement; i.e. $f_{2,3}=f_{2,4}$ and $f_{2,5}=f_{2,6}$.

Figure 4.14. Fractional Variables of Example Network with 3 Pre-Specified Links for Improvement and One of Each Intersection Type

4.2 Solution Representations and Decoding Procedures

We employ various solution representations for different metaheuristic methods. Specifically, SA, MSA and RS employ the continuous solution representations, whereas GA and RTS require binary-string solution representations. The decoding procedures are required to decode a solution representation to a vector of fractional variables (i.e. fbp for NDP, $fssd$ for SSD and $fndp_ssd$ for NDP-SSD) depending on metaheuristic algorithm.

4.2.1 Solution Representations and Decoding Procedures for NDP

Simulated Annealing Algorithm

The SA algorithm considers the dependent and independent variables. The solution representation of SA is the vector of independent variables. There is always one dependent variable, since all fraction variables are added up to one ($\sum_{i=1}^{N_{ndp_variable}} fbp'_i = 1$). For example, in Network 1 (in Figure 5.1, Chapter 5), there are five cells to be assigned budget (the sink cell is not eligible). Cells $c3$ and $c4$ belong to the same link, so they are considered as one variable. Specifically, the independent and dependent variables for Network 1 are shown in Figure 4.15.

Cell	Fractional Value of TAB	Link	Variable (fbp'_i)	Variable (fbp_i)	Number of Cells (N_fbp_i)	Independent/Dependent
c1	fb_1	1	fbp'_1	fbp_1	1	Independent
c2	fb_2	2	fbp'_2	fbp_2	1	Independent
c3	fb_3	3	fbp'_3	fbp_3	2	Independent
c4	fb_4					
c5	fb_5	4	fbp'_4	fbp_4	1	Dependent

Note: Number of independent variables, $N_{IP} = 3$

fbp'_i is the fraction of TAB assigned to link i .

fbp_i as the fraction of TAB assigned to each cell of link i .

Figure 4.15. Solution Representation of SA for NDP on Network 1 in Figure 5.1

From this example, we assume the last variable being dependent, and fbp_4 can be determined from Equation (4.2).

$$fbp_{DP} = \left(1.0 - \sum_{i=1}^{N_{IP}} (fbp_i \times N_fbp_i) \right) / N_fbp_{DP} \quad (4.2)$$

From the preliminary experimental results on NDP in Chapter 6, the SA is the least efficient as will be discussed in Section 4.3, so it is modified to improve efficiency, and called the modified SA (MSA).

Modified Simulated Algorithm

MSA views all variables as independent variables. The solution representation of MSA is the vector of real variables fr . Since the vector of fractional variables for NDP (fbp') is bounded between 0 and 1 (i.e., $fbp'_i \in [0,1] \forall i = 1, \dots, N_{ndp_variable}$), the algorithm considers fr_i as a weight for link i . It requires a decoding procedure to translate the vector fr to the vector of fractional variables fbp' and fbp for NDP as shown in Equations (4.3.1)-(4.3.2).

$$fbp'_i = \frac{fr_i}{\sum_{j=1}^{N_{ndp_variable}} (fr_j)} \quad \forall i = 1, \dots, N_{ndp_variable} \quad (4.3.1)$$

$$fbp_i = \frac{fbp'_i}{N_fbp_i} \quad \forall i = 1, \dots, N_{ndp_variable} \quad (4.3.2)$$

Genetic Algorithm

Since GA requires the binary string representation, the vectors of real numbers have to be encoded into binary string structures. For encoding procedure, the lower bound, upper bound and required precision of each decision variable must be specified. All decision variables (genes) have the same lower bound ($fr_i^{\min} = 0$) and upper bound ($fr_i^{\max} = 1$). Given the required precision ($prec$ after decimal point), the required bits (m) for each decision variable is determined from Equation (4.4) (Goldberg, 1989):

$$2^{m-1} < ((fr_i^{\max} - fr_i^{\min}) \cdot 10^{prec} + 1) \leq 2^m \quad (4.4)$$

From Equation (4.4), the required precisions of 2, 3, 4 and 5 correspond to the required bits of 7, 10, 14 and 17, respectively. Thus, the total bits required to represent a solution (i.e. the length of a chromosome) are $m \cdot N_{variable}$. Each decision variable is associated with each link to be assigned budget in the network. Furthermore, the GA implementation translates the binary structures into the packed bit arrays based on the octal number representation to maximize both space and time efficiency in manipulating structures (see further detail in Grefenstette, 1990).

Essentially, the decoding procedure (Step 1 of GA in Section 4.6) translates a binary string to the vector of fractional variables fr , since there are the boundary constraints imposed on fr (i.e. $0 = fr_i^{\min} \leq fr_i \leq fr_i^{\max} = 1 \quad \forall i = 1, 2, \dots, N_{ndp_variable}$). The decoding must transform fr to fbp' such that $\sum_{i=1}^{N_{ndp_variable}} fbp'_i = 1$ by Equations (4.3.1)-(4.3.2).

If $\sum_{i=1}^{N_{ndp_variable}} fr_i = 0$, this is called the failure of GA, and assigning arbitrary new fractional values is required to proceed (Lee, 1998). In this case, we choose to assign budget equally to all cells:

$$fbp_i = \frac{1}{\sum_{i=1}^{N_{ndp_variable}} N_fbp_i} \quad \forall i = 1, 2, \dots, N_{ndp_variable} \quad (4.5)$$

Reactive Tabu Search

Since RTS was initially invented for combinatorial optimization problems and also shown to be effective and efficient for such problems (Battiti and Tecchiolli, 1994), the appropriate solution representation of our problems for RTS should be a binary string. With such binary string representation, the problem then can be considered as a combinatorial problem; i.e., finding a best combination of 0 or 1 in all bit positions that yields the best objective function value. Specifically, the solution representation of RTS is the binary string corresponding to the vector of fractional variables fr . For each fr_i , it is bounded within the interval $[fr_i^{\min} = 0, fr_i^{\max} = 1]$. The required binary bits (m) for each fr_i is determined in the same way as in GA by Equation (4.4). Thus, a solution representation is a binary string with the length of $m \times N_{ndp_variable}$.

The decoding procedure starts by translating a binary string into a set of fractional values (fr_i); then, fbp' and fbp are determined from Equations (4.3.1)-(4.3.2). It is

possible that $\sum_{i=1}^{N_{ndp_variable}} fr_i = 0$, which corresponds to a binary string composed of zero in all bit

positions. This all-zero string will never be selected as the next solution; it is simply ignored and not considered as a valid neighbor of the current solution.

Random Search

The solution representation of RS is the vector of real variables fr . Since there are bound constraints for NDP ($fbp'_i \in [0,1] \ \forall i = 1, \dots, N_{ndp_variable}$), the algorithm considers fr_i as a weight for link i . The required decoding procedure is Equation (4.3) that translates the vector fr to the vector of fractional variables, fbp' .

4.2.2 Solution Representations and Decoding Procedures for SSD

For SSD, we consider the modified simulated annealing (MSA) algorithm instead of the original SA due to its better efficiency. MSA and RS employ the continuous solution representations, whereas GA and RTS require binary-string solution representations.

Modified Simulated Annealing Algorithm

MSA views all variables as independent variables. The solution representation of the modified SA is the vector of real variables fr . Since the vector of fractional variables for SSD, $fssd$, is bounded between 0 and 1 (i.e., $fssd_i \in [0,1] \ \forall i = 1, \dots, N_{ssd_variable}$). The algorithm considers only the fraction of the real variables fr_i , so the decoding procedure is Equation (4.6).

$$fssd_i = fr_i - \lfloor fr_i \rfloor \quad \forall i = 1, \dots, N_{ssd_variable} \quad (4.6)$$

Genetic Algorithm

The encoding procedure that encodes the vector of fractional variables $fssd$ into a binary string structure is the same as that in NDP, and the required bits, m , for each fractional variable is determined by Equation (4.4). In Chapter 6, we consider the required precision of 2 is sufficient (i.e., $m=7$) for SSD problem. The decoding procedure

that translates a binary string solution to the vector of fractional variables, $fssd$, is Equation (4.7):

$$fssd_i = fr_i = \frac{real_i}{2^m - 1} \quad \forall i = 1, 2, \dots, N_{ssd_variable} \quad (4.7)$$

where $real_i$ is a real number corresponding to the binary string associated with the i^{th} variable.

Reactive Tabu Search

The solution representation of RTS is the binary string corresponding to the vector of fractional variables, $fssd$. For each $fssd_i$, it is bounded within the interval $[fssd_i^{\min} = 0, fssd_i^{\max} = 1]$. The required binary bits (m) for each $fssd_i$ is determined in the same way as in GA by Equation (4.4). Thus, a solution representation is a binary string with the length of $m \times N_{ssd_variable}$. The decoding procedure starts by translating a binary string into a set of real numbers ($real_i$) associated to a set of variables ($fssd_i$); then, $fssd_i$ is determined from Equation 4.7.

Random Search

The solution representation of RS is the vector of fractional variables $fssd$ with $fssd_i \in [0,1] \quad \forall i = 1, \dots, N_{ssd_variable}$. Thus, it is not required a decoding procedure.

4.2.3 Solution Representations and Decoding Procedures for NDP-SSD

For NDP-SSD, we consider MSA instead of the original SA due to its better efficiency. In the same way as NDP and SSD, MSA and RS employ the continuous solution representations, whereas GA and RTS require binary-string solution representations.

Modified Simulated Annealing Algorithm

The solution representation of MSA is the vector of real variables $fr = \{fr_ndp, fr_ssd\}$ where fr corresponds to $fndp_ssd$, and fr_ndp and fr_ssd correspond to NDP and

SSD fractional variables (fbp' and $fssd$), respectively. As such, the dimension of the vector fr equals $N_{ndp-ssd_variable} = N_{ndp_variable} + N_{ssd_variable}$. The decoding procedure translates the vectors fr_ndp and fr_ssd to the vectors fbp' and $fssd$ from Equations (4.3) and (4.6), respectively.

Genetic Algorithm

The encoding procedure that encodes the vector of fractional variables $fndp_ssd = \{fbp', fssd\}$ into a binary string structure is the same as those for NDP and SSD, and the required bits (m) for each fractional variable is determined by Equation (4.4). In Chapter 6, we consider the required precision of 2 is sufficient (i.e., $m=7$) for NDP-SSD problem. The decoding procedure that translates a binary string solution to the vector of fractional variables, $fndp_ssd$, is the combination of Equations (4.3) and (4.7) (for fbp' and $fssd$, respectively).

Reactive Tabu Search

The solution representation of RTS is the binary string corresponding to the vector of fractional variables $fndp_ssd$. The encoding procedure is the same as those for NDP and SSD, and the required binary bits (m) for each fractional variable is determined in the same way as in GA by Equation (4.4). Thus, a solution representation is a binary string with the length of $m \times N_{ndp-ssd_variable}$. The decoding procedure converts a binary string to a set of real numbers ($real_i$) associated to a set of variables ($fndp_ssd_i$); then, fbp'_i and $fssd_i$ are determined from Equations (4.3) and (4.7), respectively.

Random Search

The solution representation of RS is the vector of fractional variables, $fr = \{fr_ndp, fr_ssd\}$. The decoding procedure that translates fr_ndp to fbp' is Equation (4.3), and the vector $fssd$ is simply equal to fr_ssd .

4.3 Objective Functional Evaluation

When a metaheuristic algorithm obtains a new solution (fr), the decoding procedure translates the vector fr to decision variables as discussed in Sections 4.1 and 4.2, which are application dependent. For NDP, the decision variables are the budget allocation policy to improve the capacities of pre-specified links. For SSD, the decision variables are the signal settings (cycle lengths, green splits, phase sequences and time offset) for pre-specified intersections. For NDP-SSD, the decision variables are both budget allocation policy and traffic signal settings. The objective of deterministic problem is to minimize the TSTT, whereas that of stochastic problem is to minimize the expected TSTT. The objective of robust problem is to minimize the weighted summation of the expected TSTT and the expected risk.

Specifically, for deterministic problems, the UO DTA module in VISTA (in Section 2.4.3) is called to evaluate the current solution fr , and the objective function value $z(fr)$ is set to the TSTT:

$$z(fr) = TSTT \quad \text{for deterministic problem} \quad (4.8)$$

With the current solution fr , the functional evaluation procedures of both stochastic and robust problems involve running the UO DTA for N_d times; each UO DTA run corresponds to each generated O-D demand scenario generated in the initialization step (Step 0) of the metaheuristic algorithms. The objective function values for the stochastic and robust problems are Equations (4.9)-(4.10):

$$z(fr) = \frac{\sum_{i=1}^{N_d} TSTT_i}{N_d} \quad \text{for stochastic problem} \quad (4.9)$$

$$z(fr) = \frac{\sum_{i=1}^{N_d} TSTT_i}{N_d} + \lambda \cdot \frac{\sum_{i=1}^{N_d} \max^2(0, TSTT_i - TSTT^{TARGET})}{N_d} \quad \text{for robust problem} \quad (4.10)$$

where $TSTT_i$ is the TSTT from the i^{th} run of UO DTA for $i=1, \dots, N_d$.

The next section describes the simulated annealing algorithm.

4.4 Simulated Annealing (SA)

The simulated algorithm employed in this study is derived from Lee (1998) and Vanderbilt and Louie (1984), and can be described as follows:

Simulated Annealing (after Lee, 1998; Vanderbilt and Louie, 1984)

Step 0: Set outer iteration counter $k = 1$ and inner iteration counter $m = 0$.

Initialize the initial solution: the vector of fractional values, $fr^{(0;0)}$.

Initialize the SA parameters:

- Number of inner iterations $N_{IN} \geq 15N_{IP}$;
- Number of outer iterations N_{OUT} ;
- Initial temperature, $c^{(1)} = 9.9999 \times 10^9$;
- Initial scaling matrix, $Q^{(1)} \in R^{N_{IP} \times N_{IP}}$;
- Growth factor (>1.0) for covariance determination $\chi = 2.0$;
- Boundary intensifying parameter, $\delta = 0.01$;
- Beta for covariance determination $\beta = 0.11$;
- $base = 2$ in the cooling schedule.

For stochastic and robust problems, generate N_d O-D demand scenarios (say 30) that are independent and identically distributed from their distributions (Monte Carlo sampling).

Step 1: Set $m = m+1$.

Step 2: Generate the random vector $u^{(m;k)} = (u_1^{(m;k)}, u_2^{(m;k)}, \dots, u_{N_{IP}}^{(m;k)})^T$.

Step 3: Form $\Delta fr^{(m;k)} = Q^{(k)} u^{(m;k)}$.

Step 4: Determine the new feasible point: $fr^{(m;k)} = fr^{(m-1;k)} + \Delta fr^{(m;k)}$.

Step 5: Determine the objective function, $z(fr^{(m;k)})$.

Step 6: Evaluate the new solution by the Metropolis algorithm:

If $\Delta z = z(fr^{(m;k)}) - z(fr^{(m-1;k)}) < 0$, accept $fr^{(m;k)}$.

Else generate a random number $u' \in [0,1]$

If $u' \leq \exp\left(-\frac{\Delta z}{c^{(k)}}\right)$, accept $fr^{(m;k)}$.

Else reject $fr^{(m;k)}$ and set $fr^{(m;k)} = fr^{(m-1;k)}$.

Step 7: If $m < N_{IN}$, go to Step 1. Otherwise, go to Step 8.

Step 8: If the stopping criterion is satisfied, stop. The stopping criterion (Vanderbilt and Louie, 1984) is:

$$\frac{\bar{z}^{(k)} - z_{\min}^{(k)}}{\bar{z}^{(k)}} < \varepsilon = 0.001,$$

where $\bar{z}^{(k)}$ is the average objective value of the N_{IN} steps at the k^{th} outer iterations;
and $z_{\min}^{(k)}$ is the best objective value of the N_{IN} steps at the k^{th} outer iterations.

Step 9: Form the first and second moments of the searches ($v^{(k)}$ and $w^{(k)}$, respectively):

$$v_i^{(k)} = \frac{1}{N_{IN}} \sum_{m=1}^{N_{IN}} fr_i^{(m;k)} \quad \forall i = 1, 2, \dots, N_{IP}$$

$$w_{ij}^{(k)} = \frac{1}{N_{IN}} \sum_{m=1}^{N_{IN}} \left[(fr_i^{(m;k)} - v_i^{(k)}) \cdot (fr_j^{(m;k)} - v_j^{(k)}) \right] \quad \forall i = 1, \dots, N_{IP}; j = 1, \dots, N_{IP}.$$

Form the new covariance matrix $s^{(k+1)}$: $s^{(k+1)} = \frac{\chi}{\beta \cdot N_{IN}} w^{(k)}$

Step 10: Perform the Cholesky decomposition to determine the new scaling matrix $Q^{(k+1)}$ (Lee, 1998):

Step 10.0: Set all elements of $Q^{(k+1)}$ above the main diagonal equal to zero.

Let $Q_{11}^{(k+1)} = \sqrt{s_{11}^{(k+1)}}$, form $Q_{j1}^{(k+1)} = \frac{s_{j1}^{(k+1)}}{Q_{11}^{(k+1)}} \quad \forall j = 2, 3, \dots, N_{IP}$, and set $j = 2$.

Step 10.1: Let $q_i \quad \forall i = j, j+1, \dots, N_{IP}$ be column vectors of dimension $j-1$ whose components are, respectively, the first $j-1$ elements in the i^{th} row of $Q^{(k+1)}$.

These elements are already computed: $q_i = [Q_{i,1}^{(k+1)}, \dots, Q_{i,j-1}^{(k+1)}]^T$.

Step 10.2: Form the diagonal element: $Q_{jj}^{(k+1)} = \sqrt{s_{jj}^{(k+1)} - q_j \cdot q_j}$.

Step 10.3: If $j = N_{IP}$, go to Step 10.4; otherwise compute the j^{th} column of $Q^{(k+1)}$

$$\text{below the main diagonal: } Q_{ij}^{(k+1)} = \frac{1}{Q_{jj}^{(k+1)}} (s_{ij}^{(k+1)} - q_i \cdot q_j) \quad \forall i = j+1, \dots, N_{IP}.$$

Step 10.4: If $j = N_{IP}$, stop and go to Step 11; otherwise, set $j = j+1$ and go to Step 10.1.

Step 11: Obtain the new temperature $c^{(k+1)}$ from the cooling schedule, Equation (4.11):

$$c^{(k)} = \begin{cases} base^{(|\rho|-1)} \cdot c^{(k-1)} & \text{if } \rho < 0 \\ \rho \cdot c^{(k-1)} & \text{if } \rho \geq 0 \end{cases} \quad (4.11)$$

where $base$ is positive real-valued; $\rho = \frac{\hat{z}^{(k-1)} - \hat{z}^{(k-2)}}{\hat{z}^{(k-2)}}$; and $\hat{z}^{(k-1)}$ = the minimum

z (i.e. application-dependent objective function value) of the $(k-1)^{\text{th}}$ outer iteration.

Step 12: Set $k = k+1$.

Step 13: If $k < N_{OUT}$, go to Step 1. Otherwise, terminate the algorithm.

Certain important components of the algorithm are further described.

Initialization

Although SA does not depend on an initial solution because of its global nature (Lee, 1998), in the calibration of SA parameters on Network 1 for the deterministic NDP in Section 6.2, two choices of initial solutions are tested: uniformly generated and SONDP solution. The uniformly generated initial solution is found to be better than an SONDP initial solution. Thus, for NDP, the initial solution is uniformly generated (i.e. assign equal budget to all cells). For SSD, the initial solution is randomly generated (i.e. randomly create a traffic signal setting).

One guideline for the initial scaling matrix $Q^{(l)}$ is that it should have small elements, so that most generated solutions at the initial stage are accepted (Lee, 1998). Two choices of $Q^{(l)}$ are tested in Section 6.2, Chapter 6: 1) diagonal matrix with the same elements and 2) lower triangular matrix with the same elements. For stochastic and

robust problems, the initialization step includes generating N_d O-D demand scenarios (say 30) from their distributions (Monte Carlo sampling).

With temperature $c^{(k)}$, a total of N_{IN} random searches are performed (inner iterations: Steps 1-7), followed by the outer iterations (Steps 8-13). After N_{IN} random searches, the stopping criterion is checked in Step 8. If the criterion is not met, $Q^{(k+1)}$ is determined by first finding the first and second moments of the searches in Step 9 and then performing the Cholesky decomposition in Step 10. Then, the temperature $c^{(k)}$ is reduced to $c^{(k+1)}$ by the cooling schedule in Step 11.

Inner Iterations

In Step 2, each u_i is an independent and identically distributed uniform distribution on the interval $(-\sqrt{3}, \sqrt{3})$ with zero mean and unit variance. In Step 3, the random step $(\Delta fr^{(m;k)})$ for outer iteration k and inner iteration m is determined from the product of the scaling matrix $Q^{(k)}$ and the random vector $u^{(m;k)}$. Step 4 determines the new feasible solution $fr^{(m;k)}$ that contains only independent variables, and the dependent variable must be calculated (e.g. Equation 4.2 for NDP). Since there is a bound constraint:

$$fr_i^{(m;k)} \in [0,1] \quad \forall i = 1, \dots, N_{IP}, i = DP \quad (4.12)$$

for all fractional variables, the newly generated $fr^{(m;k)}$ may not necessarily satisfy this constraint. If the bound constraint is not satisfied, the regeneration of the whole set is necessary. Lee (1998) pointed out that this method has two shortcomings: 1) when Q contains many large elements, a lot of regeneration procedures may be required and 2) the newly generated variables tend to exclude boundary values, even though boundary values are very important in optimization. He proposed a strategic approach to improve efficiency and intensify the boundary search. If a newly generated variable is just outside the boundary by a small amount δ , set this variable to the boundary value. This approach reduces the regeneration effort and searches more boundary points. However, the boundary constraint must still be held; otherwise the regeneration is required.

In Step 5, after generating new solution $fr^{(m;k)}$, the functional evaluation procedure is performed as discussed in Section 4.3. In Step 6, the algorithm calculates the change of objective value $\Delta z = z(fr^{(m;k)}) - z(fr^{(m-1;k)})$ and applies the Metropolis algorithm to decide whether $fr^{(m;k)}$ will be accepted.

Outer Iterations

After N_{IN} random iterations of Steps 1-7, the stopping criterion is checked in Step 8. In Step 9, the first and second moments of the searches are determined in order to form the new covariance matrix $s^{(k+1)}$, which in turn is used to determine the new scaling matrix $Q^{(k+1)}$ in Step 10 (Cholesky decomposition).

Step 10 is further elaborated here. The covariance matrix $s^{(k+1)}$ is symmetric, not necessarily positive definite but positive semi-definite (Lee, 1998). The generation of the vector $\Delta fr^{(m;k)}$ involves randomness resulted from the random vector $u^{(m;k)}$. As such, it is natural to expect $s^{(k+1)}$ being positive definite. According to Bronson (1989), a positive definite matrix $s^{(k+1)}$ can be factored as

$$s^{(k+1)} = Q^{(k+1)} \cdot Q^{(k+1)T}$$

where $Q^{(k+1)}$ is a lower triangular matrix with positive values on its diagonal. This decomposition can be efficiently performed by the Cholesky decomposition. Press et al. (1992) stated that the Cholesky decomposition is extremely stable numerically, and a failure simply indicates that the covariance matrix $s^{(k+1)}$ is not positive definite. Lee (1998) suggested two sources of failure: 1) accumulated numerical error due to limited machine precision and round-off as the search continues; and 2) the feasibility maintaining method for $fr^{(m;k)}$ including the approach for intensifying the boundary search. A simple check can be made to detect this failure in Step 10.2: if $s_{jj}^{(k+1)} - q_j \cdot q_j \leq 0$, $s^{(k+1)}$ is not positive definite and the decomposition fails. If the

decomposition failure is detected, set $Q^{(k+1)} = \frac{1}{1 + N_{CDF}^{(k+1)}} Q^{(1)}$, where $N_{CDF}^{(k+1)}$ is the total failures up to the $(k+1)^{th}$ outer iteration (Lee, 1998). $Q^{(l)}$ is continuously damped to

reflect the fact that the covariance becomes smaller as iterations increase. After obtaining the new scaling matrix, the next step (Step 11) is to determine the new temperature.

Step 11 involves a cooling schedule, which is further explained as follows. Two important cooling characteristics are the initial temperature and the temperature reduction procedure. The SA performance is directly related to the cooling schedule. The initial temperature should be high enough to accept most new points, and temperature reduction should be slow enough to avoid quenching and also maintain computational efficiency. Lee (1998) discussed three cooling schedules: geometric cooling, and two adaptive cooling schedules with a square root function and an exponential function. He showed that the adaptive cooling with an exponential function (Equation 4.11) is the best, so it is adopted in this study.

The simulated annealing algorithm is run until the outer iteration counter reaches N_{OUT} , and then returns the best found solution. The next section describes the modified SA, which is inspired from our preliminary experiment on the deterministic NDP in Section 6.3.2 in Chapter 6.

4.5 Modified Simulated Annealing (MSA)

Based on the preliminary results of deterministic NDP in Section 6.3.2 in Chapter 6, SA is the slowest algorithm when compared with GA and RS. From our close examination, this is mainly because of the solution representation. Specifically, it becomes more difficult to generate a new feasible solution in Step 4 as the iteration grows. Although we adopted the strategy proposed by Lee (1998) to improve the efficiency by allowing the intensification of the boundary search, the SA still appears very inefficient. In this study, we modify SA and call it the modified SA (MSA) algorithm to eliminate the cause of the infeasibility in Step 4. Instead of classifying the decision variables into independent and dependent variables, we view all decision variables as independent variables; thus, $N_{IP} = N_{variables}$. Then, the formal algorithm of MSA is similar to the original SA with the following changes:

- $N_{IP} = N_{variables}$

- The boundary intensifying parameter, δ , is deleted in Step 0.
- In Step 4, the hidden regeneration procedure due to the infeasibility of new solution is deleted, since any new generated solution will be feasible.
- The following decoding procedure is added in Step 5 to translate $fr^{(m;k)}$ into the meaningful variables before activating UO DTA:

Additional Decoding Procedure in Step 4 for MSA

For NDP, Step 4 in MSA is broken down into 2 steps:

Step 4.1: If $\sum_{i=1}^{N_{ndp_variable}} fr_i^{(m;k)} = 0$, set $fr_i^{(m;k)} = rand[0,1]$ for all $i = 1, 2, \dots, N_{ndp_variable}$.

Step 4.2: $fbp_i^{(m;k)}$ and $fbp_i^{(m;k)}$ for all $i = 1, 2, \dots, N_{ndp_variable}$ are calculated from Equations (4.3.1) and (4.3.2).

For SSD, Step 4 is the following:

Step 4: $fssd_i^{(m;k)}$ for all $i=1, 2, \dots, N_{ssd_variable}$ are determined from Equation (4.6).

For NDP-SSD, Step 4 simply combines the above-mentioned steps for NDP and SSD. The next section describes the genetic algorithm.

4.6 Genetic Algorithm (GA)

This study modifies the C source code named GENESIS Version 5.0 (Grefenstette, 1990) for the GA implementation with a modification in the functional evaluation. As discussed in Section 2.9, the GA terms that are used interchangeably are chromosome, string, individual, structure and solution. Next, let $Pop(k)$ be the binary-string population from generation k : $Pop(k) = \{string^{(1;k)}, \dots, string^{(N_{pop};k)}\}$ where $string^{(j;k)}$ represent the binary string of chromosome j of length $m \times N_{variable}$ in generation k . There are N_{POP} chromosomes per population. Let the vector $real^{(j;k)}$ represent the vector of corresponding real value of chromosome j in the population from generation k , and the fractional value $fr_i^{(j;k)} = \frac{real_i^{(j;k)}}{2^m - 1}$ for $i = 1, \dots, N_{variable}$ where m is the number of required binary bits of each decision variable. Thus, the population from iteration k can

also be written as: $Pop(k) = \{fr^{(1;k)}, \dots, fr^{(N_{POP};k)}\}$ where chromosome $fr^{(j;k)}$ is the vector of fractional variables $fr^{(j;k)} = \{fr_1^{(j;k)}, \dots, fr_{N_{variable}}^{(j;k)}\}$ for all $j = 1, \dots, N_{POP}$.

The formal algorithm of GA is described as follows.

Genetic Algorithm (after Grefenstette, 1990)

Step 0: Set iteration counter $k = 0$.

Initialize the binary-string population $Pop(0) = \{string^{(1;0)}, \dots, string^{(N_{POP};0)}\}$.

Initialize the GA parameters:

- population size;
- crossover rate;
- mutation rate;
- selection method;
- total functional evaluations (total trials)

For stochastic and robust problems, generate N_d O-D demand scenarios (say 30), which are independent and identically distributed from their distributions (Monte Carlo sampling).

Step 1: Decode all binary structures into fractional values (see Sections 4.1-4.2)

$$Pop(k) = \{fr^{(1;k)}, \dots, fr^{(N_{POP};k)}\}.$$

Step 2: Evaluate the fitness of all structures in $Pop(k)$ by two possible ways:

- With SUS proportional selection (i.e. objective function value as performance measure), for minimization problem :

$$fitness_j = \frac{Worst - z(fr^{(j;k)})}{Worst - Average_current_z} \forall j = 1, \dots, N_{POP},$$

where $Average_current_z$ is an average objective functional value in the current generation, and $Worst$ is the greatest objective functional value which has occurred in the last five generations.

- With SUS ranking selection (i.e. $z(fr^{(j;k)})$ -based ranking as performance measure)

$$fitness_j = Rank_Min + rank_j \frac{(Rank_Max - Rank_Min)}{N_{POP} - 1} \quad \forall j = 1, \dots, N_{POP}$$

where $rank_i = N_{POP} - 1$ for the best solution in the population, and
 $rank_i = 0$ for the worst.
 $Rank_Max = 2.0$ and $Rank_Min = 0.75$

Step 3: Collect the following convergence measures:

- *Online* performance = the running average of the value of the objective function of all valid individuals that have been generated,
- *Average* performance = the average of the objective function values of the individuals of each population generated,
- *Offline* performance = the average of the best objective function values,
- *Best* = the best solution found so far,
- *Conv* = Number of positions that converge (i.e. all but a few structures have the same value at that position) to a particular binary value in a generation ($Conv \in [0, Length]$),
- *Lost* = Number of positions that all structures contain the same binary value in a generation ($Lost \in [0, Length]$), and
- *Bias* = The greater fraction of $N_{POP} \times Length$ that is biased toward either 0 or 1 ($Bias \in [0.5, 1.0]$).

Step 4: If one of the following three stopping criteria is met, terminate the algorithm:

- i) number of functional evaluations (trials) is greater than or equal to the total trials,
- ii) all structures in the population are identical ($Lost = Length$), and
- iii) there is not a new generated structure in two successive generations.

Otherwise, set $k = k + 1$, and go to Step 5.

Step 5: Select $Pop(k)$ from $Pop(k-1)$ by either SUS proportional or ranking selection.

Step 6: Perform the mutation operation on $Pop(k)$.

Step 7: Perform the two-point crossover operation on $Pop(k)$.

Step 8: Go to Step 1.

Certain important components are described as follows.

In Step 0, GA parameters are initialized. The performance of GA is heavily dependent on these parameters, which will be explicitly calibrated in Chapter 6. Also, the initial population is generated in this step.

Initial Population

The initial population is randomly generated to cover a wide domain range. For the deterministic NDP in Section 6.2 in Chapter 6, two choices of the initial population are tested: 1) one SONDP solution and the others randomly generated and 2) all randomly generated. The result shows that the randomly generated initial population provides a better solution; thus, the initial populations will be randomly generated.

The chosen solution representations and established decoding procedures (Sections 4.1 and 4.2) of NDP, SSD and NDP-SSD always satisfy the constraints; i.e. for NDP, the budget constraint ($fbp'_i \in [0,1]$ for $i = 1, \dots, N_{ndp_variable}$ and $\sum_{i=1}^{N_{ndp_variable}} fbp'_i = 1$) and for SSD, the signal constraints ($fssd_i \in [0,1] \forall i = 1, \dots, N_{ssd_variable}$). Since the UO DTA is employed for functional evaluation, the UO DTA conditions are always satisfied. Thus, the constraint handling (Michalewicz, 1996), mostly based on the concept of penalty functions that penalize infeasible solutions, is not required.

After decoding in Step 1, the algorithm evaluates the fitness of all structures of the current population in Step 2.

Fitness Evaluation

In generation k , after converting the chromosomes to the vectors of fractional variables ($fr^{(k)}$), the UODTA module in VISTA is called to evaluate each chromosome (i.e. $fr^{(j;k)}$, $j = 1, 2, \dots, N_{POP}$) and determine the objective functional value, $z(fr^{(j;k)})$ as discussed in Section 4.3. The objective functional value, $z(fr^{(j;k)})$, for each solution is used to calculate a fitness measure. The determination of fitness in Step 2 for

minimization problem depends on two possible selection methods (Grefenstette, 1990): proportional and ranking. In the proportional selection, it is noted that if *Worst* is set at some great value a priori or the greatest objective functional value since the first generation, this definition of *Worst* may have the unfortunate effect of making good values of a structure hard to distinguish. In the ranking selection, *Rank_Min* and *Rank_Max* are the minimum and maximum expected values of offspring. The ranking selection algorithm is a linear mapping under which the worst structure is assigned *Rank_Min* children and the best is assigned *Rank_Max* children.

After fitness evaluation, the algorithm collects various convergence measures in Step 3, namely, *Online*, *Offline*, *Average*, *Best*, *Conv*, *Lost*, and *Bias* (Grefenstette, 1990). The descriptions of these measures are given in Step 3. In Step 4, if one of the stopping criteria (shown in Step 4) is met, the algorithm terminates. Otherwise, the algorithm proceeds to the selection process in Step 5.

Selection

A selection algorithm determines the actual number of offspring that each individual should receive based on its fitness (Baker, 1987). There are two parts in a selection algorithm: 1) determination of the individuals' expected values; and 2) conversion of the expected values to discrete numbers of offspring. The selection algorithm must maintain a constant population size while attempting to provide accurate, consistent and efficient sampling. Baker (1987) evaluated seven selection algorithms based on three measures: bias, spread, and efficiency. He found that the best selection algorithm is the stochastic universal sampling (SUS). SUS is a simple, single phase, $O(N_{POP})$ sampling algorithm. It is zero biased, has the minimum spread and will achieve all N_{POP} samples in a single traversal. The algorithm is strictly sequential as follows:

Stochastic Universal Sampling (SUS) Selection Algorithm (Baker, 1987)

ptr = rand[0,1];

For (sum = i = 0; i < N_{POP}; i++)

For (sum += expected value of individual i; sum > ptr; ptr++)

Select individual i ;

SUS guarantees that the number of offspring of any structure is bounded by the floor and by the ceiling of the real-valued expected number of offspring. SUS is analogous to a roulette wheel with N_{POP} equally spaced pointers, and each structure is allocated a portion of the roulette wheel proportional to the structure's relative fitness. A single spin of the wheel determines the number of offspring assigned to every structure, yielding N_{POP} winners. Then, the selection pointers are randomly shuffled, and the selected structures are copied into the new population.

SUS can be based on two different measures: the objective function, $z(fr^{(j;k)})$, and ranking (Grefenstette, 1990). The proportional selection simply considers the $z(fr^{(j;k)})$ as performance measure for each chromosome. For the ranking selection, the probability of selecting a structure is proportional to its index in the population, where the index of the best structure is $N_{POP}-1$, and the index of the worst structure is 0. The ranking selection helps prevent premature convergence by preventing super individuals from taking over the population within a few generations. However, the ranking selection often produces slower improvement than the proportional selection. Note that there is a generation gap parameter that indicates the fraction of the population which is replaced in each generation. In this study, we choose the generation gap of one.

Furthermore, DeJong (1975) proposed the elitism concept in his GA. The elitist policy stipulates that the best individual always survives into the new generation. The elite individual is placed in the last position in the offspring population and is not changed through crossover or mutation. In the calibration of GA parameters in Section 6.2 in Chapter 6, we consider four choices of selection algorithm as shown: 1) SUS proportion and elitism, 2) SUS proportion without elitism, 3) SUS ranking and elitism, and 4) SUS ranking without elitism. After the selection process, the algorithm performs the mutation in Step 6.

Mutation

After the new population is selected, the mutation is applied to the new population. The mutation operator helps GA search different neighborhoods. Each position is given a chance of M_rate to undergo mutation. M_rate is usually set less than 0.1 in either a fixed or a variable rate (Lee, 1998). This study employs a fixed M_rate . Mutation is implemented by treating the entire population as a linear array of positions. For each position, generate a random number between 0 and 1. If the random number is within M_rate , a random value is chosen from $\{0,1\}$, and the value of the position is overwritten by this random value. The next step (Step 7) is the crossover operation.

Crossover

A crossover operator exchanges genetic materials among adjacent pairs of the first $(C_rate \times N_{POP})$ structures in the new population. It is recalled that the population is already randomly shuffled in the selection procedure. The crossover can be implemented in a variety of ways, but there are theoretical advantages that the two-point crossover is better than other kinds (DeJong, 1975). The two-point crossover operator treats the structures as rings, randomly chooses two crossover points, and exchanges the sections between these points (Grefenstette, 1990). The segments between the crossover points are exchanged if the parents differ outside of the crossed segment. After crossover, if the offspring are different from the parents, then the offspring replace the parents. It is noted that we use the same random seed for all UO DTA runs; thus, UO DTA is not a noisy function, so structures that are identical to parents will not be marked for evaluation. An example of the two-point crossover implemented in this study is shown in Figure 4.16.

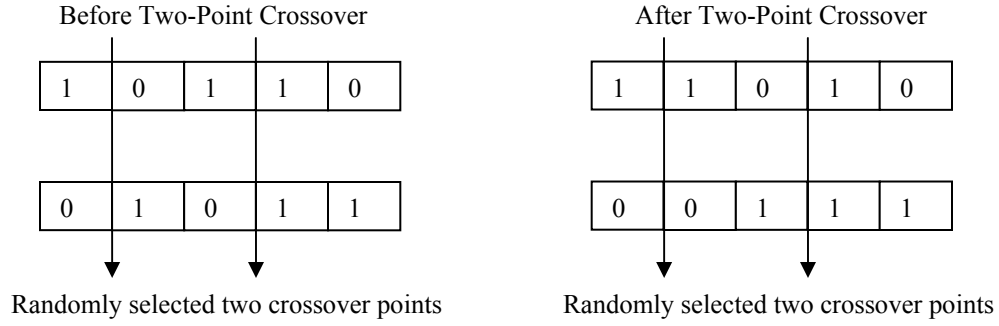


Figure 4.16. Example of Two-Point Crossover

After the crossover operation, the algorithm iterates Steps 1-8 until one of the stopping criteria is met in Step 4. The next section describes the reactive tabu search algorithm.

4.7 Reactive Tabu Search (RTS)

RTS maintains the basic building blocks of tabu search developed by Glover (1989, 1990). That is, RTS employs a set of temporarily forbidden moves, and the duration of prohibition is controlled by the tabu tenure or tabu list size. There are two novel search mechanisms introduced by Battiti and Tecchiolli (1994) in RTS. Firstly, they added a fully automated procedure of adapting the tabu tenure to the problem and to the current evolution of the search. Secondly, they introduced an escape strategy to diversify the search when the search believably is confined in a chaotic attractor.

As discussed in Sections 4.2, the solution representation of RTS is a binary string that requires a decoding procedure to convert it to decision variables. Certain notations in RTS are first given as follows. The record *current* describes the status of the search by storing the current binary-string solution in *current.config[]*, the associated objective functional value in *current.f*, the current iteration (each iteration consists of neighborhood evaluations and move selection) in *current.time*, and the current number of functional evaluations until the current time in *current.feval*. The algorithm keeps track of the best found solution in the similar record *best_so_far*. The variable *list_size* is the tabu tenure that dynamically shrinks or expands based on the evidence of the search. The array

latest_time_negate[bit_position] stores the time (iteration number) that the bit position *bit_position* is negated (i.e. changed from 0 to 1, or 1 to 0). This array is used to check if a move is tabu. The variable *chaotic* is the number of often repeated configurations, and *moving_average* is the moving average value of the cycle length. The variable *pointer* stores the address of the location of the record in the hashing structure. A record in the hashing structure includes *last_time* (the most recent iteration number that the configuration is found), *repetitions* and *config[]*. Other variables are self-described. The formal algorithm of RTS is formally described below.

Reactive Tabu Search (Battiti and Tecchiolli, 1994)

Step 0: Initialization

- Initialize the data structures for hashing.
- Initialize the data structures for tabu:
 - Set $\text{BITS_PER_SOLUTION} = \text{required number of bits } (m) \times N_{\text{variable}}$.
 - Set *latest_time_negate[bit_position]* = -infinity, for *bit_position* = 0 to BITS_PER_SOLUTION
 - Set *list_size* = 1; *chaotic* = 0; *moving_average* = 0;
steps_since_last_size_change = 0; *current.time* = 0; *current.feval* = 0.
- Randomly generate an initial binary-string configuration (*current.config[]*)
- Set *current.feval* equal to the initial objective functional value
- Initialize the *best_so_far* record, and set *best_so_far.f* = *current.f*
- For stochastic and robust problems, generate N_d O-D demand scenarios (say 30), which are independent and identically distributed from their distributions (Monte Carlo sampling).

Step 1: Activate the procedure *Check_for_repetitions(current.config)*:

Step 1.1: Set *steps_since_last_size_change* = *steps_since_last_size_change* + 1.

Step 1.2: Search for the *current.config[]* in the hashing structure

Step 1.3: If the configuration is found, go to Step 1.4.

Else if the configuration is not found,

- Install the configuration in the hashing structure
(i.e. store *current.config* in *pointer->config*)
- Set *pointer->last_time = current.time*; *pointer->repetitions = 1*.
- Go to Step 1.7.

Step 1.4: Set *pointer* = the location of the record.

Set *Length = current.time - pointer->last_time*.

Set *pointer->last_time = current.time*; (*pointer->repetitions*)++.

Go to Step 1.5.

Step 1.5: If *pointer->repetitions* is greater than REP, perform the following:

- Add the current configuration to the set of often-repeated configurations (*repeated_config[chaotic][]*).
- Set *chaotic = chaotic + 1*.
- If *chaotic > CHAOS*, set *chaotic = 0* and go to Step 6 (return ESCAPE).

Otherwise go to Step 1.6.

Step 1.6: If *Length < CYCLE_MAX*, perform the following:

- Set *moving_average = 0.1×Length + 0.9×moving_average*.
- Set *list_size = list_size×INCREASE*.
- Set *steps_since_last_size_change = 0*.

Go to Step 1.7.

Step 1.7: If *steps_since_last_size_change > moving_average*, do the following:

- Set *list_size = max(1, list_size×DECREASE)*.
- Set *steps_since_last_size_change = 0*.

Go to Step 2 (return DO_NOT_ESCAPE).

Step 2: Activate the procedure *Choose_best_move* (DO_NOT_ESCAPE from Step 1.7):

Step 2.1: Set *time_past = current.time - list_size*;

Set *aspirated_or_not_tabu = tabu_and_not_aspirated = FALSE*;

If *move_type = 0*, set $N_{neighbor} = m \times N_{variable}$.

Else if *move_type = 1 or 2*, set $N_{neighbor} = N_{variable}$.

Step 2.2: Evaluate the neighbors of the current configuration by iterating the following steps for $N_{neighbor}$ iterations.

Step 2.2.1: Obtain a neighbor (depend on move types) and associated $chosen_bit_array[]$ that stores bit positions to be negated.

Step 2.2.2: Evaluate the neighbor by calling UO DTA (see Section 4.3).
Set $current.feval = current.feval + 1$.

Step 2.2.3: If the neighbor is not found in the hashing structure, install it.

Step 2.2.4: Check if this move is prohibited by activating $is_tabu(chosen_bit_array)$:

- If $move_type = 1$ or 0 , then if $latest_time_negate[chosen_bit_array[0]] \geq time_past$, this move is tabu.
- Else if $move_type = 2$, then if $latest_time_negate[chosen_bit_array[i]] \geq time_past$ for all $i=1, 2, \dots, N_{variables}$, this move is tabu.

Step 2.2.5: If $is_tabu(chosen_bit_array)$ indicates the move is tabu, then activate the procedure $aspiration_criterion$:

- If the objective functional value of the neighbor is better than $best_so_far.f$, this move is aspirated.

Step 2.2.6: If the neighbor is aspirated or not tabu, set $aspirated_or_not_tabu = TRUE$.

Else set $tabu_and_not_aspirated = TRUE$.

Step 2.2.7: If the objective value of the neighbor is better than the best neighbor, set the best neighbor equal to this neighbor.

Step 2.3: If $aspirated_or_not_tabu = TRUE$, go to Step 3. Otherwise, Step 2.4.

Step 2.4: If $tabu_and_not_aspirated = TRUE$, set $list_size = \max(1, list_size \times DECREASE)$ and go to Step 3.

Step 3: Activate the procedure $make_tabu(chosen_bit_array)$:

- If $move_type = 1$ or 0 , set $latest_time_negate[chosen_bit_array[0]] = current.time$.
- Else if $move_type = 2$, set $latest_time_negate[chosen_bit_array[i]] = current.time$, for $i = 1$ to $N_{variable}$.

Step 4: Negate the binary value(s) of $current.config[]$ according to $chosen_bit_array[]$ (the number of chosen bit positions depends on $move_type$).

Step 5: Activate the procedure *update_current_and_best*:

- Set $current.time = current.time + 1$.
- Set $current.f$ equal to the functional value of the best neighbor.
- If $current.f$ is better than $best_so_far.f$, update $best_so_far$.

Go to Step 7.

Step 6: Enter the diversification phase (ESCAPE from Step 1.5) by the procedure *escape*:

Step 6.1: Delete the hashing memory structure and set $chaotic = 0$.

Step 6.2: Generate a random number of steps in the given range by setting

$$Steps = 1 + 0.5 \times (1 + rand[0,1]) \times moving_average.$$

Step 6.3: Perform the following process for $Steps$ iterations:

- If $move_type = 0$ or 1 , randomly select a bit position and assign it to $chosen_bit_array[0]$. Then, negate the corresponding binary value of $current.config[]$.
- Else if $move_type = 2$, randomly selected a bit position within substring j and assign it to $chosen_bit_array[j]$, for $j = 1$ to $N_{variable}$. Then, negate the corresponding binary values of $current.config[]$.
- Evaluate the new solution by calling UO DTA (see Section 4.3);
($current.feval$)++
- Set $current.f$ equal to the new objective functional value.
- Activate the procedure *make_tabu(chosen_bit_array)* (see Step 3).
- Activate the procedure *update_current_and_best* (see Step 5).

Step 7: If $current.feval > max_iterations$, terminate. Otherwise, go to Step 1.

The reactive tabu search (RTS) starts off with initialization, followed by the main loop of the search that continues to be executed until a maximum number of iterations (*max_iterations*) is reached. The initialization (Step 0) includes randomly generating an initial solution, initializing RTS parameters and allocating memory for the open hashing table. In the main loop (Steps 1-7), it first compares the current solution with the previously visited solutions stored in memory by calling the procedure *Check_for_repetitions*. This procedure returns two possible values: DO_NOT_ESCAPE or ESCAPE. If the returned value is DO_NOT_ESCAPE, the algorithm goes to Step 2. In Step 2, the algorithm activates the procedure *Choose_best_move* to choose the best next move from the current solution based on a selected neighborhood definition (that will be discussed in the next sub-section). If the returned value is ESCAPE, it is believed that the search trajectory remains confined in a limited region of the space, and the algorithm enters a diversification phase by activating the procedure *escape* (Step 6). In the main loop, the algorithm also keeps track of the best solution found so far. It is noted that Battiti and Tecchiolli (1994) claims that RTS does not require parameter calibration because RTS is originally designed to be deterministic, and the algorithm is learning while searching. The important components of RTS are further described as follows.

Neighborhood definitions

Three neighborhood definitions and associated move types are considered, namely types 0, 1 and 2. For neighborhood type 0, a neighbor is a configuration that is the same as the current configuration except one bit. Thus, the number of neighbors equals the number of bits in a configuration (i.e. $N_{neighbor} = m \times N_{variable}$ where m is the required number of bits per variable). To illustrate, Figure 4.17 shows a binary string of the vector of three variables and $m = 3$. There are nine type-0 neighbors as shown in Figure 4.18a. Since there is not a closed form formula or an efficient algorithm for move evaluation, each neighbor of the current solution is evaluated by running the simulation-based UO DTA (see Section 4.3), which is very computationally expensive for stochastic and robust problems. With move type 0, for this simple example for deterministic

problems, it takes nine DTA runs to choose the best move from the current solution to the next, so within a limit of CPU time the search trajectory is shorter than the other move types and may cover fewer areas of solution space. This leads to the introduction of move types 1 and 2 that are stochastic neighborhood definition. For move types 1 and 2, we consider the deterministic number of neighbors equals to the number of variables ($N_{neighbor} = N_{variable}$), so that presumably the search can move through the solution space faster than move type 0. Move type 1 considers a binary sub-string corresponding to a variable. It randomly selects a bit position within this sub-string, and negates the bit value. This is iterated for each sub-string of each variable, yielding the number of neighbors equal to the number of variables. Move type 2 considers all sub-strings for all variables at once. It randomly selects a bit position for each sub-string, and negates the bit values. This is iterated for $N_{neighbor}$ iterations. These type-1 and type-2 neighborhood definitions are illustrated in Figures 4.18b and 4.18c.

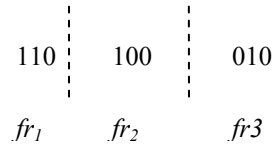


Figure 4.17. Binary-String Solution Representation
(3 variables and the required bits of 3)

<u>0</u> 10	100	010		110	<u>0</u> 00	010		110	100	<u>1</u> 10
1 <u>0</u> 0	100	010		110	1 <u>1</u> 0	010		110	100	<u>0</u> 00
11 <u>1</u>	100	010		110	10 <u>1</u>	010		110	100	01 <u>1</u>

a) Nine Deterministic Neighbors of the Solution in Figure 4.17 based on Move Type 0

↓ 11 <u>1</u>	100	010		110	↓ <u>0</u> 00	010		110	100	↓ 01 <u>1</u>
------------------	-----	-----	--	-----	------------------	-----	--	-----	-----	------------------

Note: An arrow indicates a randomly selected bit position within a binary sub-string associated with a variable.

b) Three Stochastic Neighbors of the Solution in Figure 4.17 based on Move Type 1

↓ <u>0</u> 10	↓ 10 <u>1</u>	↓ <u>1</u> 10		↓ 1 <u>0</u> 0	↓ <u>0</u> 00	↓ 01 <u>1</u>		↓ 11 <u>1</u>	↓ <u>0</u> 00	↓ <u>0</u> 00
------------------	------------------	------------------	--	-------------------	------------------	------------------	--	------------------	------------------	------------------

Note: An arrow indicates a randomly selected bit position within a binary sub-string associated with a variable.

c) Three Stochastic Neighbors of the Solution in Figure 4.17 based on Move Type 2

Figure 4.18. Example of Neighborhood Definition based on Move Types 0, 1 and 2

Objective Functional Evaluation

The objective functional value, $current.f$, is determined differently for deterministic, stochastic and robust problems, and the procedures are as discussed in Section 4.3. These procedures are also employed in the initialization step (Steps 0), the neighborhood evaluation (Step 2.2.2), and the escape procedure (Step 6.3)

Memory structures

The variables in the formal algorithm and memory structures are further described as follows. For move types 0 and 1, the index of chosen bit position is stored in

chosen_bit_array[0], the bit position that will be subjected to the negation. For move type 2, the indices of chosen bit positions are stored in *chosen_bit_array[i]* for $i = 1, 2, \dots, N_{variable}$. All visited points in the configuration space are stored in the records (in the hashing structure) that contain the binary string configuration (*config[]*), the most recent time when it was encountered (*last_time*), and its multiplicity (*repetitions*). When the number of repetitions for a given point is greater than REP (=3), the configuration is added to the set of often-repeated ones (*repeated_config[chaotic][]*). The constants INCREASE (=1.1) and DECREASE (=0.9) determine the amounts by which the *list_size* is increased in the fast reaction mechanism, or decreased in the long-term size reduction procedure. The variable *moving_average* denotes a moving average of the detected cycle length, and the variable *steps_since_last_size_change* denotes the number of iterations executed after the last change of list size. These two variables are used for the long-term size reduction. The variable *chaotic* counts the number of often-repeated configurations whose value of CHAOS (=3) triggers the search to enter the diversification phase (i.e. execute the procedure *escape*). The maximum number of iterations *max_iterations* are used for terminating the search.

Short-Term and Long-Term Memory-Based Reactive Mechanisms

The memory-based reactive mechanisms of the algorithm modify the search trajectory, so that limit cycles and confinements in limited regions of the search space do not arise (Battiti and Tecchiolli, 1994). The reaction is based on the past history of the search, and it causes possible changes to the tabu list size *list_size*.

In Step 1, if the current configuration is found to be repeated, three possible changes to *list_size* can be triggered. First, the short-term-memory reactive mechanism in Step 1.6 discourages short limit cycles by increasing *list_size* by the factor INCREASE (=1.1) if the cycle length (i.e., *Length* = the current time minus the last visit time) is less than or equal to CYCLE_MAX (=50). Second, the algorithm checks that if the number of repetitions is greater than the threshold REP (=3) in Step 1.5, this repeated solution is stored in the set of repeatedly found configurations. If the size of this set is greater than

the threshold CHAOS ($=3$), the algorithm activates the long-term-memory diversification mechanism. In other words, the diversification procedure is activated as soon as more than CHAOS solutions, which are configurations that are repeated REP times during the search, are found (Battiti and Tecchiolli, 1994). Third, after a phase of the search with many repetitions, *list_size* becomes large. If the algorithm does not allow the decrease of *list_size*, the tabu tenure will remain large even in later phases when a smaller value is sufficient to avoid short limit cycles. Thus, the long-term-memory reactive mechanism in Step 1.7 executes a reduction by the factor DECREASE ($= 0.9$) if the tabu tenure remains unchanged for a number of iterations greater than *moving_average*.

In Step 2.4, if all moves from the current solutions are tabu and the functional values of all neighbors after tabu moves are not better than the best solution found so far, this implies the search is too constrained by the current list size. Thus, the long-term-memory reactive mechanism decreases the current list size by the factor DECREASE ($=0.9$).

Diversification Phase

The procedure *escape* in Step 6 executes a number of random movements (*Steps*) whose duration is proportional to the moving average of detected cycles (*moving_average*) (Battiti and Tecchiolli, 1994). The number of random movement, *Steps*, is sufficient to obtain a considerably different solution. The rationale is that longer average cycles are evidence of a larger basin, so more escape steps are likely to be required. For each step of a random walk, the algorithm randomly selects a neighbor from the selected neighborhood definition (i.e. it is not necessarily the best neighbor), and the corresponding move becomes tabu in order to discourage the search from immediately returning to the old region after completing the diversification phase. In addition, for each diversification phase, the open hashing memory is reset (i.e. all stored configurations are deleted), and the number of repeated found configurations is reset to 0 (set *chaotic* = 0).

Three Basic Tabu Search Procedures

Three basic procedures of tabu search are *is_tabu* (Step 2.2.4), *aspiration_criterion* (Step 2.2.5), *make_tabu* (Step 3). *is_tabu* and *aspiration_criterion* are routinely activated when the algorithm evaluates neighbors of the current configuration. The procedure *make_tabu* is called before the algorithm executes the best move to obtain the new configuration in Step 3 and after each escaping move in Step 6.3.

The procedure *is_tabu* depends on the move types. For move types 0 and 1, a move is tabu if it selects the bit position that has been chosen within the latest *list_size* iterations. For move type 2, a move is tabu if it selects a set of bit positions that has been chosen within the latest *list_size*. The procedure *make_tabu* stores the current time in the array *latest_time_negate[]* when a move is executed. As such, checking the tabu status of a move requires only small CPU time since it only requires a comparison of the current time with the latest occupation time in *latest_time_negate[]* (Battiti and Tecchiolli, 1994). The aspiration criterion is satisfied if the functional value after a tabu move is better than the best solution found so far.

Details on Hashing

Battiti and Tecchiolli (1994) presented certain competitive structures and algorithms for storing and retrieving the information about the history of the search process in a fast way. The entries of the search history that should be stored are the configurations, the corresponding functional values, and the times (iteration numbers) when they were encountered. The various schemes differ in their time-space complexity and in the amount of data stored per iteration. Battiti and Tecchiolli (1994) employed the hashing technique, which is standard in computer science (Aho et al., 1985). The basic idea of hashing is that the entries of search history are stored in buckets whose indices are obtained in a scattered way from the entries by using the hashing function. In our implementation, we use the hashing function that takes the functional value of configuration as its argument. Specifically, the hash function is:

$$\text{Hash}(\text{current.config}) = (\text{current.f} \times 100) \text{ modulo HASH_TABLE_SIZE},$$

where `HASH_TABLE_SIZE` is 100,000 (default value).

The search time is approximately constant and equal to a very small number of machine cycles if the number of buckets is large enough (Battiti and Tecchiolli, 1994). If the number of buckets is very large such that different entries end up in different buckets, then, only a comparison with a single stored entry is sufficient. When more than one entry is stored in the same bucket, this situation is called a collision. One way to deal with collisions is to associate a list of entries to each bucket that is enlarged when new entries arrive. This is the open hashing technique based on the array of bucket table headers, which contain the pointer to the first entry in the associated list. From Figure 4.19, *bucket[Hash(config)]->config* stores the first configuration in the list, *bucket[Hash(config)]->last_time* stores the last time when it was encountered, and *bucket[Hash(config)]->repetitions* stores the number of repetitions. Entries in the list are chained with the pointer *next* that gives the address of the next entry. The pointer *nil* is used for list termination. The number of buckets (`HASH_TABLE_SIZE`) has to be larger, so that there are fewer numbers of collisions. A rule of thumb is to make it about at least two times the maximum number of stored entries, assuming that the hashing function scatters the entries in an almost uniform manner (Battiti and Tecchiolli, 1994). The next section describes the random search algorithm.

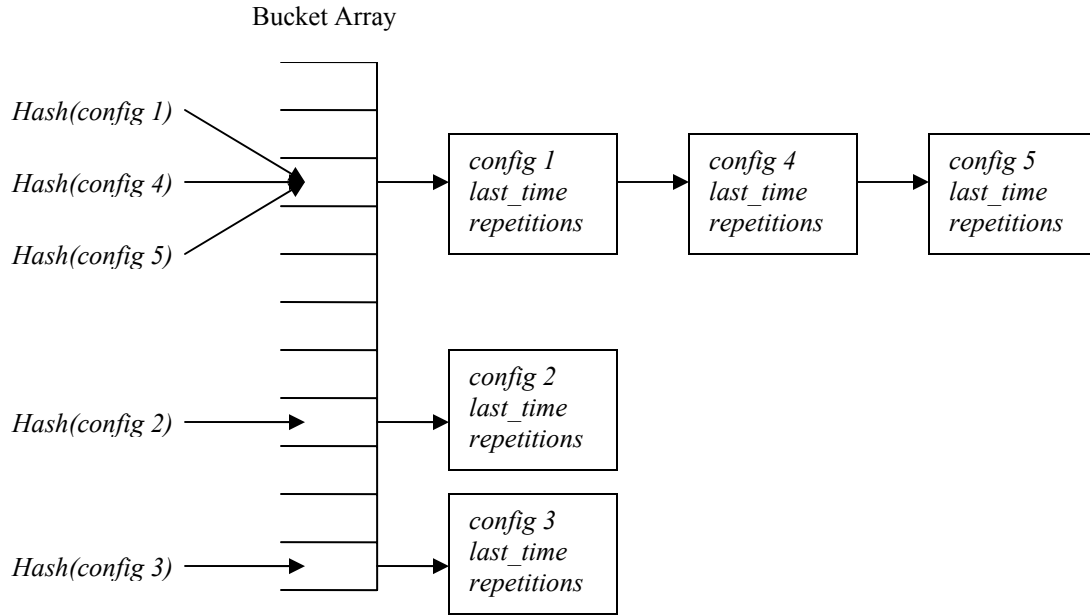


Figure 4.19. Memory Structure for the Open Hashing Scheme
(After Battiti and Tecchiolli, 1994)

4.8 Random Search (RS)

The formal random search algorithm is described as follows.

Random Search Algorithm (after Brooks, 1958)

Step 0: Set iteration counter $k = 1$; set $Best = +\infty$; and initialize $total_trials$. For stochastic and robust problems, Monte Carlo sample N_d O-D demand scenarios (say 30), which are independent and identically distributed from their distributions.

Step 1: Set $fr_i^k = rand[0,1]$, for all $i=1,2,\dots,N_{variable}$.

Step 2: Decode fr_i^k for all $i=1,2,\dots,N_{variable}$ to obtain decision variables.

(see Sections 4.1-4.2)

Step 3: With the solution derived from the vector fr^k , determine the objective functional value, $z(fr^k)$. (see Section 4.3)

Step 4: If $z(fr^k) < Best$, set $Best = z(fr^k)$ and $fr^{Best} = fr^k$.

Step 5: If $k = total_trials$, stop. Otherwise, set $k = k+1$ and go to Step 1.

The solution representation of RS and the decoding procedures are discussed in Section 4.2. There is only one RS parameter (i.e. *total_trials*), so no parameter calibration is required unlike SA, MSA and GA. After the initialization step (Step 0), Step 1 randomly generates a random number in the range between 0 and 1 associated with each decision variables. Based on these random numbers, the decision variables are determined as follows.

For NDP, Step 2 is broken down into 2 steps:

Step 2.1: If $fr_i^k = 0$ for all $i=1,2,\dots,N_{ndp_variable}$, then go to Step 1.

Step 2.2: fbp'_i and fbp_i are determined from Equations (4.3.1) and (4.3.2).

For SSD, Step 2 is the following:

Step 2: Set $fssd_i^k = fr_i^k$ for all $i=1,2,\dots,N_{ssd_variable}$.

For NDP-SSD, Step 2 simply combines Step 2 of NDP and SSD.

4.9 Summary

Five metaheuristic methods are proposed in this chapter, namely, simulated annealing (SA), modified simulated annealing (MSA), genetic algorithm (GA), reactive tabu search (RTS) and random search (RS). For clarity, the decision variables of network design problem (NDP), DTA-based signal setting design problem (SSD) and the combined NDP, SSD and DTA problem (NDP-SSD) are first specified. Then, the solution representations and decoding procedures with respect to different metaheuristics are described. Next, the objective functional evaluation procedures for deterministic, stochastic and robust optimizations are discussed. Subsequently, the formal algorithms of the five metaheuristic methods are presented. In the next chapter, we show the computational experiences for different analytical models.

Chapter 5. Experimental Results of Analytical Approach

In Chapter 3, several mathematical models are developed, and exact solution methods are proposed for network design problem (NDP), DTA-based signal setting design problem (SSD) and the combined NDP, SSD and DTA problem (NDP-SSD). In this chapter, we show the computational experience of the proposed bi-level mathematical models on two single-destination cell transmission networks. The experimental results of NDP models are first discussed, including the preliminary experiment of the deterministic NDP models, the comparison of the three new NDP models and the existing NDP models, and the comparison of the robust NDP models based on moving and fixed targets. Then, the experimental results of NDP-SSD models are shown, including the comparison of the proposed six NDP-SSD models and the existing NDP-SSD model, and the comparison of the sequential NDP and SSD approach against the integrated NDP-SSD approach.

5.1 Experimental Results of NDP Models

We consider a test problem, which is a multi-origin single-destination cell transmission network. This simple network helps us intuitively understand how the network performance will vary when designing the network improvement based on different optimization models. The LP and MIP models in Sections 5.1.2.2 and 5.1.3 are implemented in GAMS/CPLEX and MINLP in GAMS/DICOPT on a machine with Windows OS, Pentium 4-M CPU, 1.80 GHz and 640 MB of RAM. We first describe a cell transmission test network, followed by the preliminary results of the deterministic NDP models (SONDP, UONDP and BLPNDP), the comparison of all seven NDP models, and the comparison of the robust NDP models based on moving and fixed targets.

5.1.1 Cell Transmission Test Network (Network 1)

Figure 5.1 shows Network 1, composed of 6 cells and 6 cell connectors, and associated data (jam density, saturation flow and network-improvement unit costs). All cells except the sink cell are considered for link expansion. The simulation period is eight time intervals. As reviewed in Section 2.4.2.2, the vector M_t in Figure 5.1c satisfies the following inequality to enforce the UO behavior (Ukkusuri, 2002):

$$M_t - M_{t-1} > (M_{|T|} - M_t)TD$$

where TD is the total demands. For Network 1, the M_t values for time intervals $t1$ and $t2$ can be set at any value because it is impossible for a vehicle to arrive earlier than time interval $t3$.

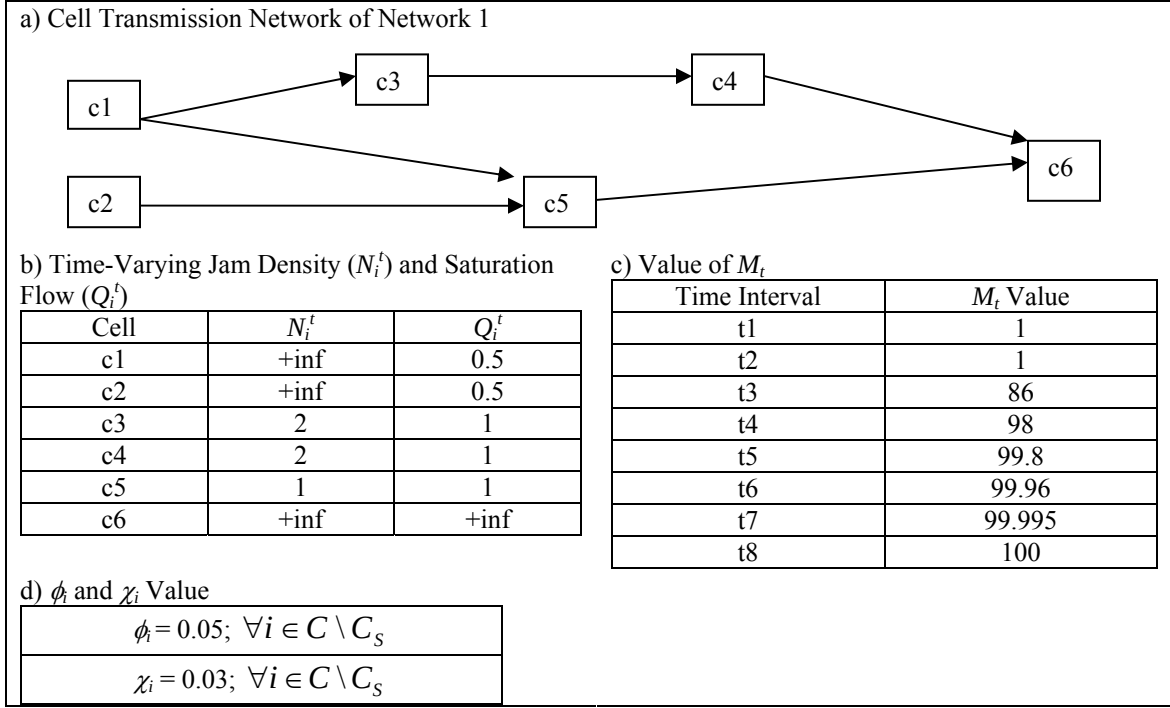


Figure 5.1. Test Network 1

5.1.2 Preliminary Results of Deterministic NDP Models

The preliminary experiment is conducted for the three deterministic NDP models (SONDP, UONDP and BLPNDP) using Network 1 in Figure 5.1 and the time-dependent demands in Figure 5.2.

Cell	t1	t2
c1	2	1
c2	2	1

Note: All other cells and time intervals have zero demand.

Figure 5.2. Time-Dependent Demand (d_i^t)

The K^{th} best algorithm and the KKT-based MIP reformulation are tested in the next two sub-sections, respectively.

5.1.2.1 K^{th} Best Algorithm

From our initial experience of the K^{th} best algorithm, the degeneracy leads to extremely long computational time, so we modified the K^{th} -best algorithm by ignoring

the existence of degeneracy (see Section 3.4.1.1). This leads to a heuristic with a deterministic error bound. The modified K^{th} -best algorithm is implemented in C++ and run on a Dell Precision 530 Workstations with dual 1.8 GHz Xeon processors and 1 GB of RAM, running under SuSE Linux 9.0. The subproblem (UO DTA) is solved by ILOG/CPLEX 9.0 Concert Technology Library, and the matrix manipulations of the simplex tableau are performed by GNU Scientific Library (GSL) 1.5, a numerical library under the GNU General Public License (Linear Algebra, Combinations and Matrices). This algorithm is tested on Network 1 with the deterministic demands in Figure 5.2. The BLPNDP results are compared with the SONDP and UONDP, shown in Figure 5.3. The deterministic error bounds of BLPNDP on Network 1 can be determined from Figure 5.3c; i.e., the deterministic error bound equals $\text{UO_TSTT}(\text{BLPNDP})$ minus $\text{SO_TSTT}(\text{SONDP})$, where $\text{UO_TSTT}(\text{BLPNDP})$ is the total system travel time (TSTT) of BLPNDP solution under UO DTA condition, and $\text{SO_TSTT}(\text{SONDP})$ is the TSTT of SONDP solution under SO DTA condition.

Based on the modified K^{th} best algorithm, a possible heuristic for multi-destination problem can be developed as discussed in Section 3.4.1.1. However, in the possible heuristic, a move from a current solution requires the generation of all adjacent extreme points from the simplex tableau of multi-destination SONDP and the associated matrix manipulation of the large simplex tableau. This is considered inefficient, and at best, it will yield only a local optimal solution. Thus, we rather adopt metaheuristics in Chapter 4 to solve multi-destination larger-size problems, since they are relatively easier to implement, have less computational effort in obtaining a move from a current solution, are widely used in the literature, and can potentially find a global optimum.

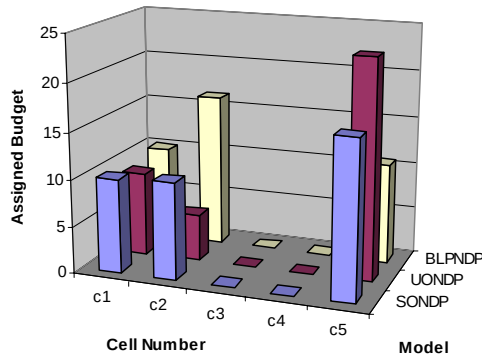
In the next section, another solution method is used to verify if the BLPNDP solutions in Figure 5.3 are globally optimal.

a) Comparison of SONDP, UONDP and BLPNDP on Network 1

TAB	SONDP, UONDP, BLPNDP			
	$b1$	$b2$	$b5$	UO TSTT
37	10, 8.875, 10	10.33, 5.000, 16.375	16.67, 23.125, 10.625	22.93, 24.7, 22.09
38	10, 9.25, 10	11.33, 5.000, 16.750	16.67, 23.750, 11.250	22.73, 24.55, 21.98
39	10, 9.625, 10	12.33, 5.000, 17.125	16.67, 24.375, 11.875	22.53, 24.40, 21.86
40	10, 10, 10	13.33, 5.000, 17.500	16.67, 25.000, 12.500	22.33, 24.25, 21.75
41	10, 10, 10	14.33, 5.375, 17.875	16.67, 25.625, 13.125	22.13, 24.08, 21.64
42	10, 10, 10	15.33, 5.750, 18.250	16.67, 26.250, 13.750	21.93, 23.91, 21.53
43	10, 10, 10	16.33, 6.125, 18.625	16.67, 26.875, 14.375	21.73, 23.74, 21.41
44	10, 10, 10	17.33, 6.500, 19.000	16.67, 27.500, 15.000	21.53, 23.58, 21.30
45	10, 10, 10	18.33, 6.875, 19.375	16.67, 28.125, 15.625	21.33, 23.41, 21.19
46	10, 10, 10	19.33, 7.250, 19.750	16.67, 28.750, 16.250	21.13, 23.24, 21.08

Note: $b3 = b4 = 0$ for all cases

b) Budget Allocations (Total Budget = 37)



c) Relationships between TSTT and Total Budget

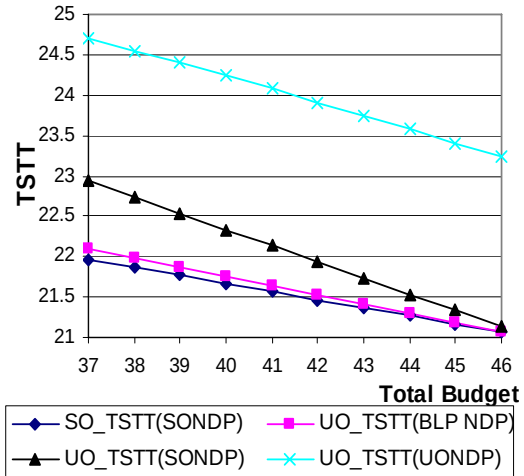


Figure 5.3. Comparison of SONDP, UONDP and BLPNDP on Network 1 with Deterministic Demands in Figure 5.2

5.1.2.2 KKT-Based MIP Reformulation

As discussed in Section 3.4.1.2, the KKT-based MIP reformulation is implemented in GAMS/CPLEX. The solutions from this approach are found to be the same as those from the modified K^{th} best algorithm for all budget values. Thus, these solutions in Figure 5.3 will be used as a benchmark for calibrating the SA, MSA and GA parameters in Section 6.2 in Chapter 6. Interestingly, from Figures 5.3a and 5.3b, the

budget allocations from SONDP, UONDP and BLPNDP are significantly different, and the TSTT obtained from the BLPNDP solutions are the lowest across all budget values as portrayed in Figure 5.3c. Thus, the bi-level NDP model does provide better solutions than the other single-level models. The next section describes the results of comparing all seven NDP models.

5.1.3 Comparison of Seven NDP Models

The experiment is conducted to compare the proposed NDP models (BLPNDP, SLP2-BLPNDP and RO-BLPNDP) and the existing NDP models (SONDP, UONDP, SLP2-SONDP, SLP2-UONDP) using Network 1 in Figure 5.1 and the stochastic time-dependent demands in Figure 5.4. For the deterministic NDP models, the mean values of time-dependent demands in Figure 5.4 are employed.

Cell	t1	t2
c1	2, 0.5	1, 0.5
c2	0.5, 2	0.5, 1

Note: 1. All other cells and time intervals have zero demand.
2. The two scenarios are equally likely ($p^{\omega 1} = p^{\omega 2} = 0.5$).

Figure 5.4. Stochastic Time-Dependent Demand ($d_i^{t,\omega 1}$, $d_i^{t,\omega 2}$)

The seven NDP models are solved to obtain exact solutions using the exact solution methods discussed in Section 3.4. The single-level NDP LP models (SONDP, UONDP, SLP2-SONDP and SLP2-UONDP) are coded in GAMS and solved by CPLEX. The linear bi-level NDP models (BLPNDP and SLP2-BLPNDP) are transformed to equivalent single-level MIPs that are then solved by GAMS/CPLEX. The robust bi-level NDP (RO-BLPNDP) model is transformed to an equivalent single-level MINLP, which contains all linear terms except the convex quadratic objective function. This is solved by GAMS/DICOPT to obtain a global optimum. For RO-BLPNDP, since we do not have information on the preferred expected TSTT of the system, we choose to set $TSTT^{TARGET}$ equal to the optimal expected TSTT from SLP2-BLPNDP. Furthermore, two values of

the goal programming weight (i.e. $\lambda = 10$ and 100) are used in the robust NDP model. The detailed results are shown in Appendix C.

In the SLP2-SONDP and SLP2-UONDP models, the time-varying demands are stochastic; these mean the parameters in the right-hand side of the constraints are random. That is, in the standard form (Equation 2.15) $\tilde{\xi} = \text{vec}(\tilde{d}, \tilde{B})$. As discussed in Section 2.5, $h(x, \tilde{\xi})$ is then convex on the support of $\tilde{\xi} (\Xi_{\tilde{\xi}})$, and Jensen's inequality results in $Eh(x, \tilde{\xi}) \geq h(x, E\tilde{\xi})$. Consequently, the SONDP and UONDP models employing the mean values of demands yield underestimated solutions (in terms of their objective functions) to SLP2-SONDP and SLP2-UONDP models, respectively. That is, the SONDP solution yields TSTT less than the expected TSTT from SLP2-SONDP as shown in Table 5.1. The UONDP model yields its objective value $(\sum_{(i,j) \in E_S} \sum_{t \in T} (M_t \cdot y_{ij}^t))$ less than

$\sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot M_t \cdot y_{ij}^{t,\omega})$ of SLP2-UONDP, but the UONDP solution may not necessarily yield less TSTT than the expected TSTT of SLP2-UONDP solutions. From our experiment, Table 5.1 shows that the UONDP does underestimate TSTT when compared with $E[TSTT]$ of SLP2-UONDP.

For the bi-level problem, Jensen's inequality cannot be guaranteed. Let's consider the recourse function of SLP2-BLPNDP, and denote by $hp(b, \tilde{\xi}^\omega)$ the nested program, which is the same as the recourse function of SLP2-UONDP. By applying Jensen's inequality to the nested program, $hp(b, E\tilde{\xi}) \leq E[hp(b, \tilde{\xi})]$. Note that

$$hp(b, E\tilde{\xi}) = \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t*} \text{ where } x_i^{t*}, y_{ij}^{t*} \text{ are optimal solution to the nested program for}$$

the mean value of demands, and $E[hp(b, \tilde{\xi})] = \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} p^\omega M_t y_{ij}^{t,\omega*}$ where $x_i^{t,\omega*}, y_{ij}^{t,\omega*}$

are optimal solution to the nested program for demand scenario ω . As

such, $\sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t*} \leq \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} p^\omega M_t y_{ij}^{t,\omega*}$; this does not necessarily imply

that $\sum_{(i,j) \in E_S} \sum_{t \in T} t \cdot y_{ij}^{t*} \leq \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} p^\omega \cdot t \cdot y_{ij}^{t,\omega*}$. Therefore, the recourse function of SLP2-

BLPNDP may not be convex on the support of $\tilde{\xi}$ and Jensen's inequality may not be guaranteed. The BLPNDP model employing the mean values of demands does not necessarily yield underestimated solutions to the SLP2-BLPNDP. However, the results of our experiment in Table 5.1 show that the BLPNDP model with the mean values of demands underestimates the TSTT when compared with the expected TSTT from SLP2-BLPNDP at the budget range between 37 and 46.

Table 5.1. TSTT and $E[TSTT]$ of Deterministic and Two-Stage Stochastic NDP models

TAB	TSTT of SONDP	$E[TSTT]$ of SLP2-SONDP	TSTT of UONDP	$E[TSTT]$ of SLP2-UONDP	TSTT of BLPNDP	$E[TSTT]$ of SLP2-BLPNDP
37	12.362	13.717	12.612	13.732	12.3625	13.475
38	12.325	13.617	12.575	13.677	12.325	13.4
39	12.287	13.517	12.537	13.622	12.2875	13.325
40	12.25	13.417	12.5	13.568	12.25	13.25
41	12.212	13.317	12.462	13.514	12.2125	13.175
42	12.175	13.217	12.425	13.459	12.175	13.1
43	12.137	13.117	12.387	13.405	12.1375	13.025
44	12.1	13.017	12.35	13.35	12.1	12.95
45	12.062	12.917	12.312	13.296	12.0625	12.875
46	12.025	12.817	12.275	13.241	12.025	12.8

Note: 1. SONDP, UONDP and BLPNDP employ mean values of demands.

2. SLP2-SONDP, SLP2-UONDP and SLP2-BLPNDP employ stochastic demands.

3. TSTT of SONDP and $E[TSTT]$ of SLP2-SONDP are under the SODTA condition.

4. TSTT of UONDP and BLPNDP, and $E[TSTT]$ of SLP2-UONDP and SLP2-BLPNDP are under the UODTA condition.

Next, for a fair comparison, the solutions of the seven models at ten total available budget (TAB) levels are evaluated under the stochastic condition to obtain TSTTs for the two demand scenarios ($TSTT^{\omega 1}$ and $TSTT^{\omega 2}$). Then, the expected TSTT ($E[UO_TSTT]$), the expected risk ($E[Risk]$) and the expected robust objective ($E[Robust_Obj(\lambda)]$) are calculated from Equations (5.1)-(5.3):

$$E[UO_TSTT] = \sum_{\omega \in \Omega} (p^\omega \cdot TSTT^\omega)$$

(5.1)

$$E[Risk] = \sum_{\omega \in \Omega} p^\omega \cdot (\max(0, TSTT^\omega - TSTT^{TARGET}))^2 \quad (5.2)$$

$$E[Robust_obj(\lambda)] = E[UO_TSTT] + \lambda \times E[Risk] \quad (5.3)$$

For our test network, $\Omega = \{\omega_1, \omega_2\}$ with $p^{\omega_1} = p^{\omega_2} = 0.5$. Figure 5.5 shows the expected TSTT of all NDP solutions, and Figure 5.6 depicts the expected risks.

As can be seen from Figures 5.5a and 5.6a, the SONDP, UONDP, BLPNDP, SLP2-SONDP, and SLP2-UONDP yield worse expected TSTTs and worse expected risks than the SLP2-BLPNDP and RO-BLPNDP. From Figures 5.5b and 5.6b, the SLP2-BLPNDP yields the lowest expected TSTTs over the budget range, but it gives higher expected risks when compared with RO-BLPNDP solutions. The lower degree of robustness in RO-BLPNDP implies that the model views the risk term ($E[Risk]$) less importantly; that is, the expected TSTT is more favored. The RO-BLPNDP($\lambda=100$) yields the least expected risks across the budget range as shown in Figure 5.6b; the RO-BLPNDP($\lambda=10$) and SLP2-BLPNDP the second and third least expected risks. Recall that SLP2-BLPNDP is simply the RO-BLPNDP with $\lambda=0$.

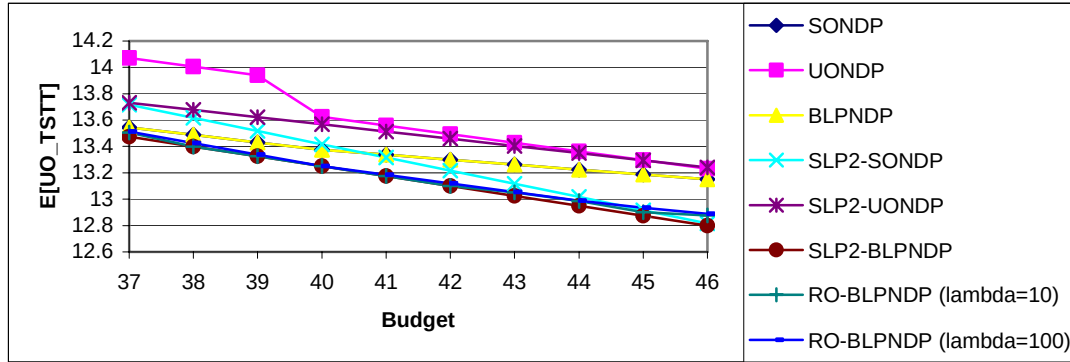
Next, we evaluate the expected robust objective of all solutions with $\lambda=10$ and $\lambda=100$ over the budget range [37, 46]. Figures 5.7a-5.7d show the results, and Figure 5.7e provides the rankings of all NDP solutions. The SLP2-UONDP and UONDP solutions perform the worst; the SONDP and BLPNDP solutions the second worst; and the SLP2-SONDP solutions the third worst. Figures 5.7a-5.7b show that RO-BLPNDP yields the most robust solution, and no other model provides a satisfactory solution across the budget range. This shows the worthiness of RO-BLPNDP.

Moreover, to illustrate the solution robustness and the worthiness of RO-BLPNDP, the risks of different NDP solutions are depicted for the two demand scenarios at the total available budget of 43 in Figure 5.8a. When the slope of the line in Figure 5.8a is steeper, the TSTTs of the two scenarios are more wildly different. The SLP2-UONDP yields the steepest line in Figure 5.8a; SONDP and BLPNDP the second; and UONDP the third. RO-BLPNDP ($\lambda=100$) yields the flattest line; RO-BLPNDP ($\lambda=10$) the second; and SLP2-BLPNDP the third. This reiterates the virtue of RO-BLPNDP.

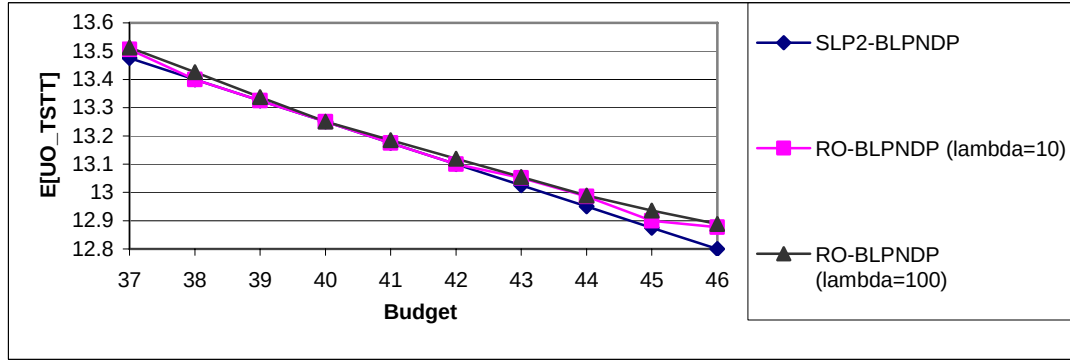
Next, we illustrate how the change of weight (or degree of robustness), which represents trade-off between the expected TSTT and the expected risk, plays role in the

robust network design. Figures 5.8c-5.8d show the effects of different weight values on the expected TSTT and the expected risk at the total budget of 43. When the weight is increased from zero to a critical weight value, the expected TSTT rapidly increases, while the expected risk drastically decreases. After this critical weight value, the expected TSTT and the expected risk gradually change with the increase of the weight. This kind of visualized graph can be utilized by the transportation planners in order to learn their preferences on the value of the goal programming weight by comparing the expected TSTT and the expected risk on the system at different weight values. Further, Figure 5.8b shows the risks of RO-BLPNDP solutions with different weight values for all demand scenarios with the budget of 43. It shows that the higher weight values are associated with the flatter line; that is, the more robust solutions are also less sensitive to the variation of O-D demand. Finally, we show the budget allocation policies based on different NDP models at the total budget of 43 in Figure 5.8e. Apparently, different NDP models recommend different policies except the SONDP and BLPNDP models, which provide the same solution for this particular test network.

a) Graph of $E[UO_TSTT]$ of All NDP Solutions



b) Graph of $E[UO_TSTT]$ of SLP2-BLPNDP, RO-BLPNDP($\lambda=10$ and $\lambda=100$)

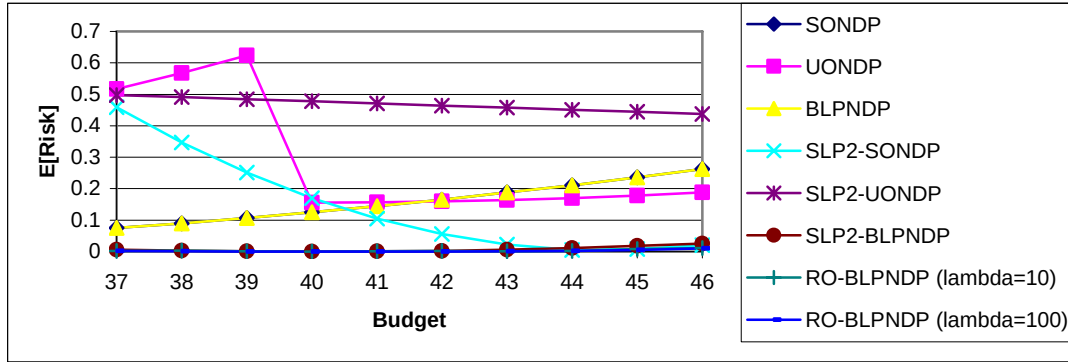


c) Numerical Values of $E[UO_TSTT]$ of Different NDP Solutions at Different Budget Levels

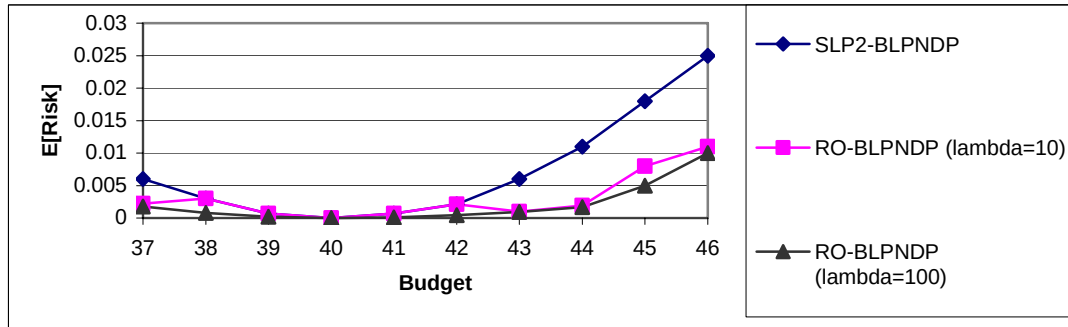
TAB	NDP Models							
	SONDP	UONDP	BLPNDP	SLP2-SONDP	SLP2-UONDP	SLP2-BLPNDP	RO-BLPNDP ($\lambda=10$)	RO-BLPNDP ($\lambda=100$)
37	13.5438	14.0719	13.5438	13.7167	13.7318	13.475	13.5056	13.5131
38	13.4875	14.0063	13.4875	13.6167	13.6773	13.4	13.4	13.4252
39	13.4313	13.9407	13.4313	13.5167	13.6227	13.325	13.325	13.3372
40	13.375	13.625	13.375	13.4167	13.5682	13.25	13.25	13.25
41	13.3375	13.5594	13.3375	13.3167	13.5136	13.175	13.175	13.1845
42	13.3	13.4938	13.3	13.2167	13.4591	13.1	13.1	13.1194
43	13.2625	13.4281	13.2625	13.1167	13.4045	13.025	13.0509	13.0543
44	13.225	13.3625	13.225	13.0167	13.35	12.95	12.9858	12.9892
45	13.1875	13.2969	13.1875	12.9167	13.2955	12.875	12.9	12.9356
46	13.15	13.2313	13.15	12.8167	13.2409	12.8	12.8769	12.8878

Figure 5.5. Expected Total System Travel Time of Different NDP Solutions at 10 Budget Levels

a) Graph of $E[\text{Risk}]$ of All NDP Solutions



b) Graph of $E[\text{Risk}]$ of SLP2-BLPNDP, RO-BLPNDP($\lambda=10$ and $\lambda=100$)



c) Numerical Values of $E[\text{Risk}]$ of Different NDP Solutions at Different Budget Levels

TAB (Target TSTT)	SONDP	UONDP	BLPNDP	SLP2- SONDP	SLP2- UONDP	SLP2- BLPNDP	RO- BLPNDP ($\lambda=10$)	RO- BLPNDP ($\lambda=100$)
37 (13.475)	0.075	0.517	0.075	0.459	0.498	0.006	0.00224	0.00175
38 (13.4)	0.09	0.568	0.09	0.347	0.491	0.003	0.003	0.000779
39 (13.325)	0.107	0.624	0.107	0.251	0.484	0.0007031	0.0007031	0.000197
40 (13.25)	0.125	0.156	0.125	0.17	0.478	0	0	0
41 (13.175)	0.144	0.157	0.144	0.105	0.471	0.000703	0.000703	0.000108
42 (13.1)	0.165	0.16	0.165	0.056	0.464	0.002112	0.002112	0.000426
43 (13.025)	0.188	0.164	0.188	0.022	0.458	0.006	0.001	0.000957
44 (12.95)	0.211	0.17	0.211	0.005	0.451	0.011	0.00192	0.001698
45 (12.875)	0.236	0.178	0.236	0.008	0.445	0.018	0.008	0.005
46 (12.8)	0.263	0.188	0.263	0.02	0.438	0.025	0.011	0.01

Figure 5.6. Expected Risks of Different NDP Solutions at 10 Budget Levels

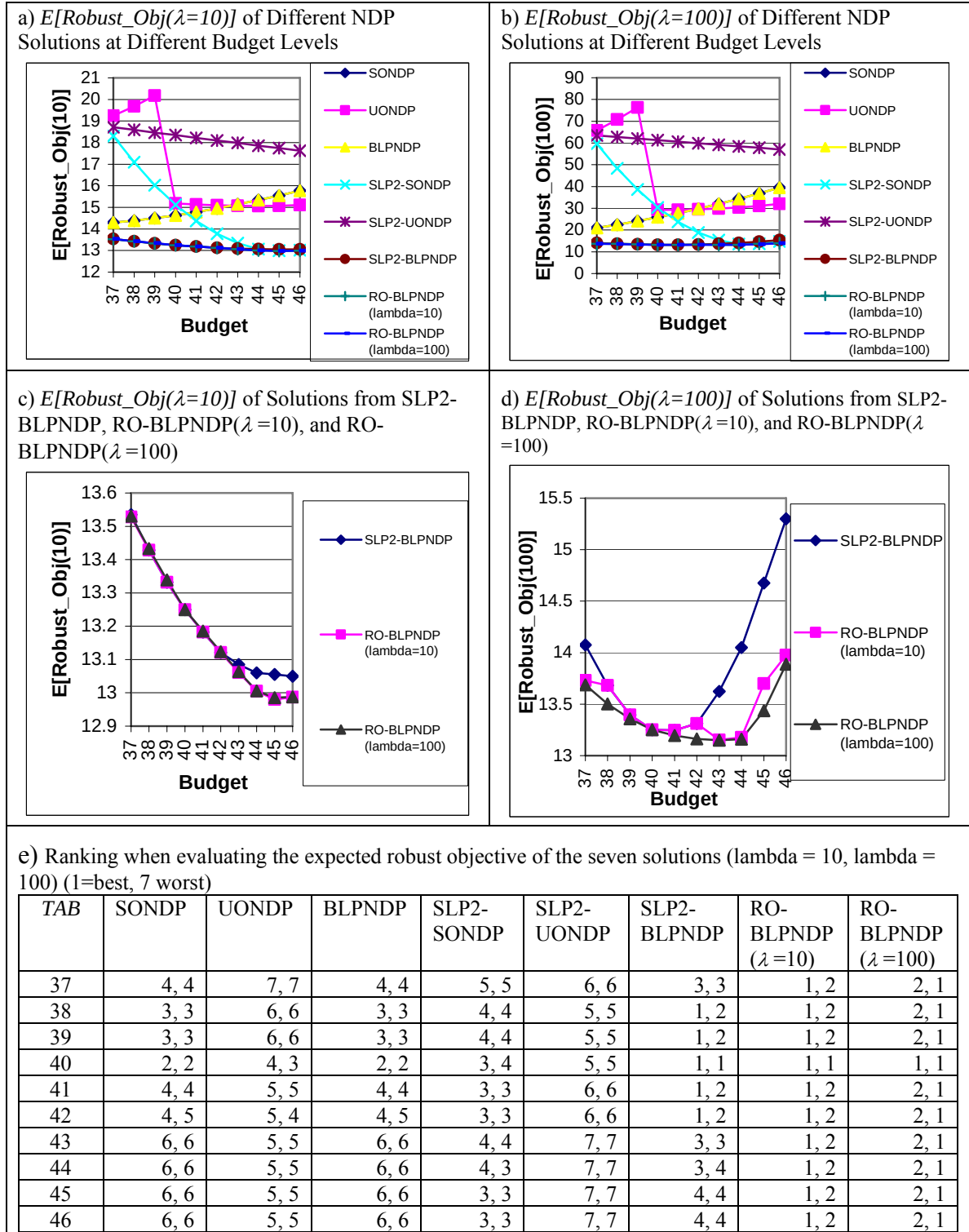


Figure 5.7. Robustness Evaluation with $\lambda = 10$ and 100 of All NDP Solutions at Ten Total Available Budget Levels

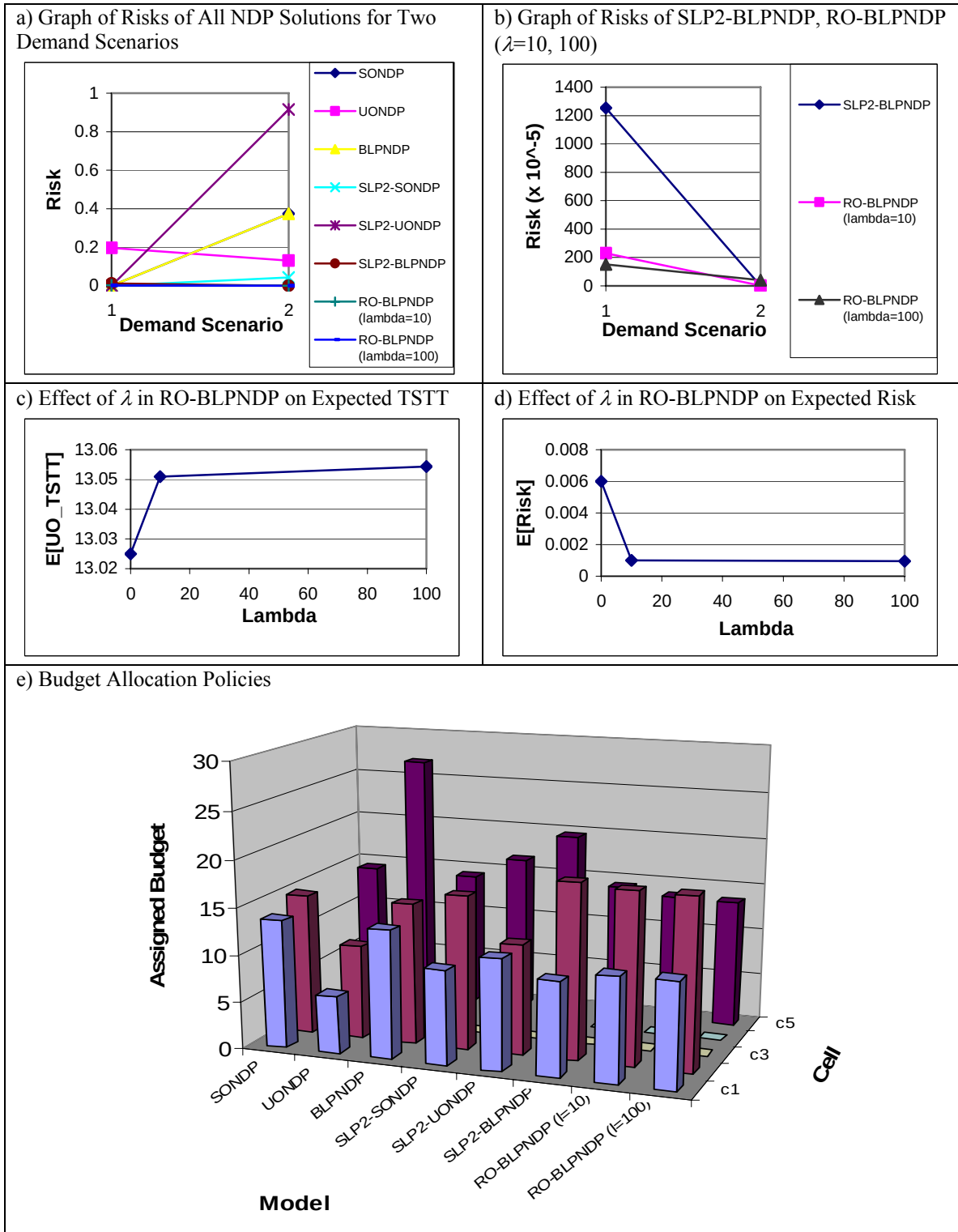


Figure 5.8. Exploration of Different NDP Solutions at Total Available Budget of 43 ($TSTT^{TARGET} = 13.025$)

5.1.4 Comparison of Robust NDP Models Based on Moving and Fixed Target TSTT

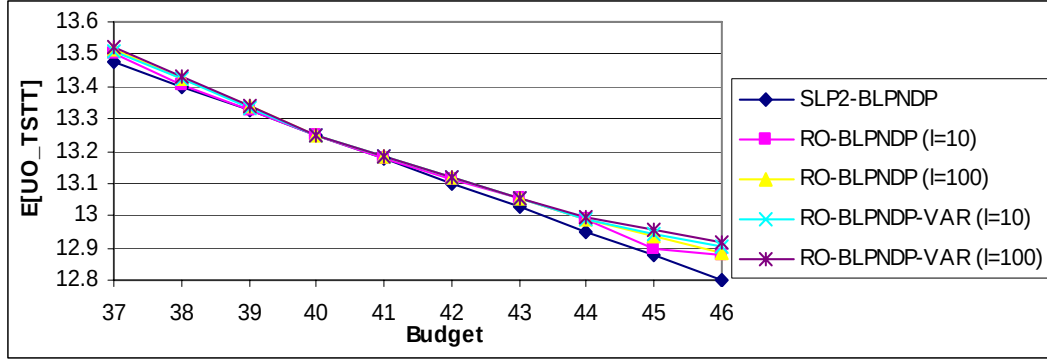
We discuss various objective functions of the robust optimization model in Chapter 3, and conclude that the fixed target formulation that penalizes only the up-side deviation should be desirable. In this section, we explore an alternative of the robust optimization model that minimizes the weighted summation of the $E[UO_TSTT]$ and the variance of TSTT ($V[UO_TSTT]$). We focus on only NDP for simplicity, and use Network 1 in Figure 5.1 and the stochastic time-dependent demands in Figure 5.4. Also, we denote the alternative robust NDP formulation by RO-BLPNDP-VAR. The exact solution method for RO-BLPNDP-VAR is the same as RO-BLPNDP. We solve SLP2-BLPNDP, RO-BLPNDP($\lambda=10, 100$) and RO-BLPNDP-VAR($\lambda=10, 100$) on Network 1 for 10 total budget values. Then, these solutions are evaluated under the stochastic condition to obtain TSTTs for the two demand scenarios, and $E[UO_TSTT]$, $E[Risk]$ and $E[Robust_Obj(\lambda=10, 100)]$ are computed from Equations (5.1)-(5.3). And $E[Robust_Obj_Var(\lambda=10, 100)]$ are determined from Equation 5.4:

$$E[Robust_Obj_Var(\lambda)] = E[UO_TSTT] + \lambda \times V[UO_TSTT] \quad (5.4)$$

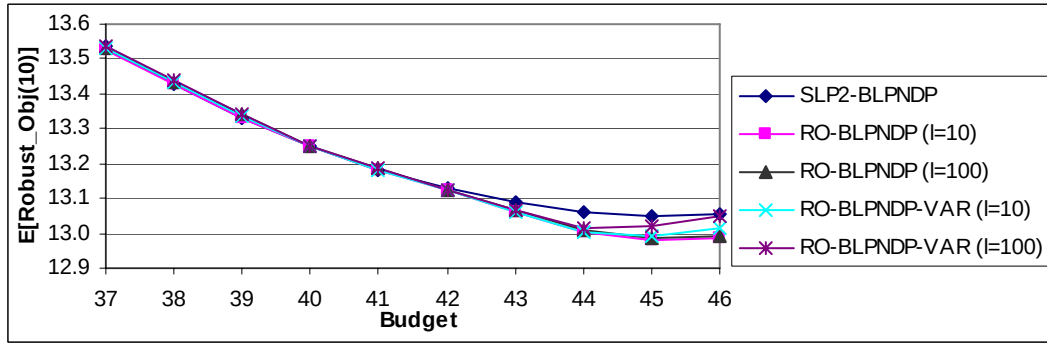
The detailed solutions and evaluation results are shown in Appendix D. Figures 5.9a-5.9e show the evaluation results in terms of $E[UO_TSTT]$, $E[Robust_Obj(\lambda=10, 100)]$ and $E[Robust_Obj_Var(\lambda=10, 100)]$ over the budget range. As expected, SLP2-BLPNDP, RO-BLPNDP($\lambda=10, 100$) and RO-BLPNDP-VAR($\lambda=10, 100$) perform the best over the budget range in terms of $E[UO_TSTT]$, $E[Robust_Obj(\lambda=10, 100)]$ and $E[Robust_Obj_Var(\lambda=10, 100)]$, respectively, since these measures are their respective objective functions. Then, with $TAB=43$, the evaluation results in terms of $E[UO_TSTT]$, $E[Risk]$ and $V[UO_TSTT]$ are portrayed in Figure 5.10. SLP2-BLPNDP gives the least $E[UO_TSTT]$ but with the worst $E[Risk]$ and $V[TSTT]$. RO-BLPNDP-VAR($\lambda=100$) yields less $V[UO_TSTT]$ and greater $E[UO_TSTT]$ than RO-BLPNDP-VAR($\lambda=10$) as RO-BLPNDP-VAR($\lambda=100$) put the higher goal programming weight on the variance term. RO-BLPNDP-VAR($\lambda=10$ and 100) yield higher $E[Risk]$ than RO-BLPNDP($\lambda=100$) since RO-BLPNDP-VAR does not incorporate the risk term in its

objective function. In addition, the risk values of the two demand scenarios are plotted in Figure 5.11. Apparently, the flattest line in Figure 5.11 is RO-BLPNDP-VAR($\lambda=100$) as its objective function includes the variance term. It is noted that the transportation planners will decide on which of the two robust NDP models (RO-BLPNDP versus RO-BLPNDP-VAR) is more desirable. If the transportation planners want to have a control on the variance of TSTT, RO-BLPNDP-VAR should be chosen. If the planners emphasize on the TSTT being less than the fixed target TSTT, RO-BLPNDP is needed.

a) Graph of $E[UO_TSTT]$



b) Graph of $E[Robust_Obj(\lambda=10)]$



c) Graph of $E[Robust_Obj(\lambda=100)]$

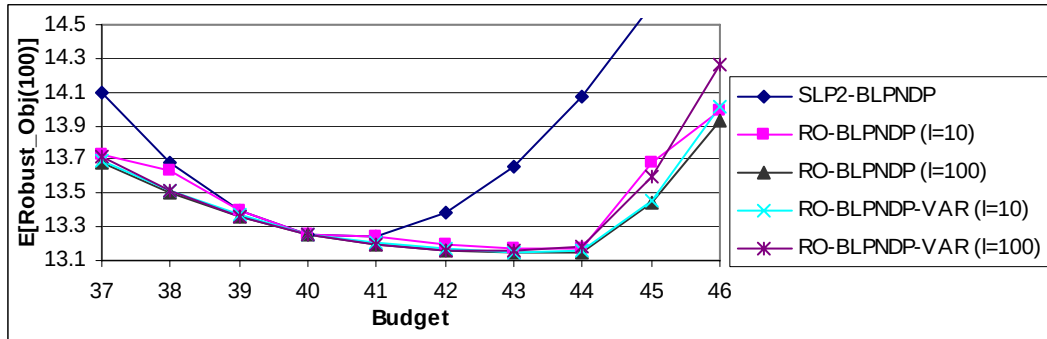
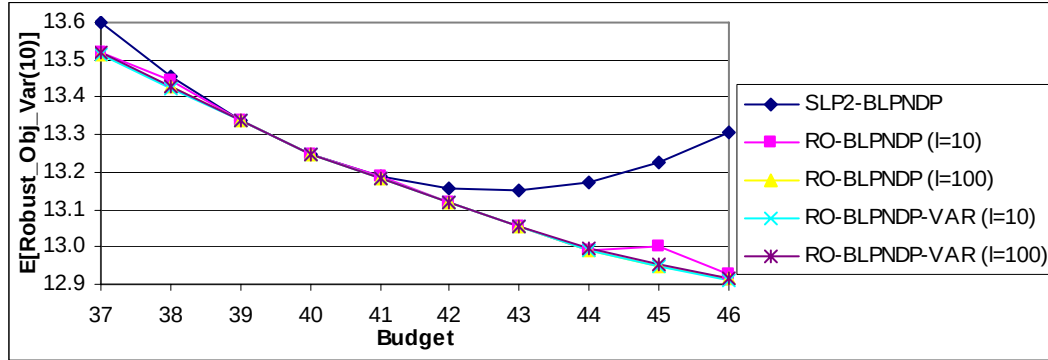


Figure 5.9 $E[UO_TSTT]$, $E[Robust_Obj(\lambda=10, 100)]$ and $E[Robust_Obj_Var(\lambda=10, 100)]$ of SLP2-BLPNDP, RO-BLPNDP($\lambda=10, 100$) and RO-BLPNDP-VAR($\lambda=10, 100$) Solutions over the Budget Range [37, 46]

d) Graph of $E[Robust_Obj_Var(\lambda=10)]$



e) Graph of $E[Robust_Obj_Var(\lambda=100)]$

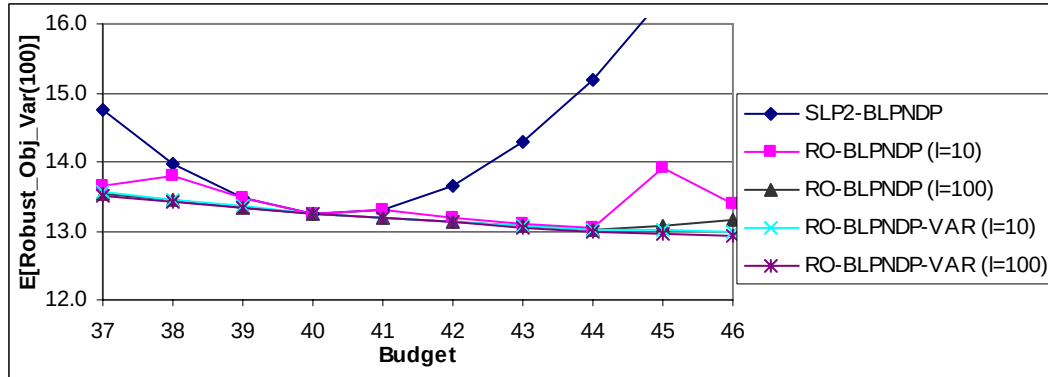
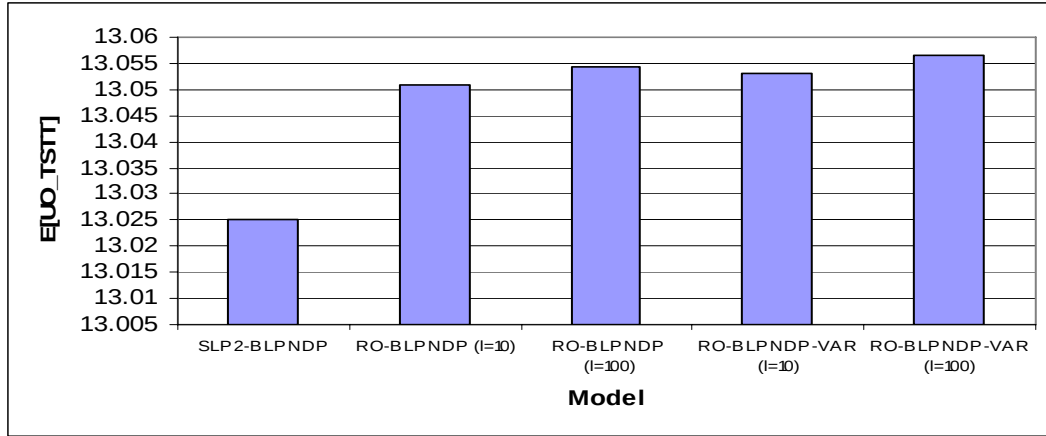
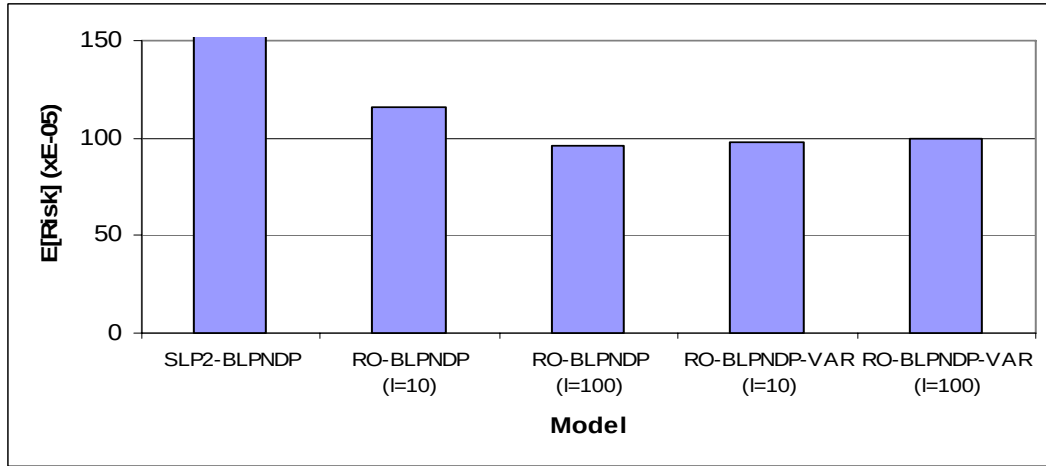


Figure 5.9 $E[UO_TSTT]$, $E[Robust_Obj(\lambda=10, 100)]$ and $E[Robust_Obj_Var(\lambda=10, 100)]$ of SLP2-BLPNDP, RO-BLPNDP($\lambda=10, 100$) and RO-BLPNDP-VAR($\lambda=10, 100$) Solutions over the Budget Range [37, 46] (cont'd)

a) Graph of $E[UO_TSTT]$



b) Graph of $E[Risk]$



c) Graph of $V[UO_TSTT]$

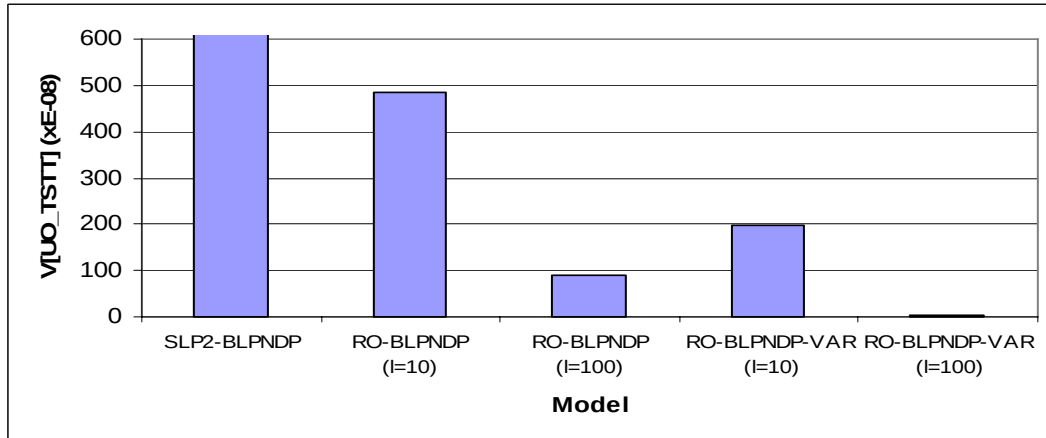


Figure 5.10. $E[UO_TSTT]$, $E[Risk]$ and $V[TSTT]$ of SLP2-BLPNDP, RO-BLPNDP($\lambda=10, 100$) and RO-BLPNDP-VAR($\lambda=10, 100$) Solutions (Budget = 43 and $TSTT^{TARGET}=13.025$)

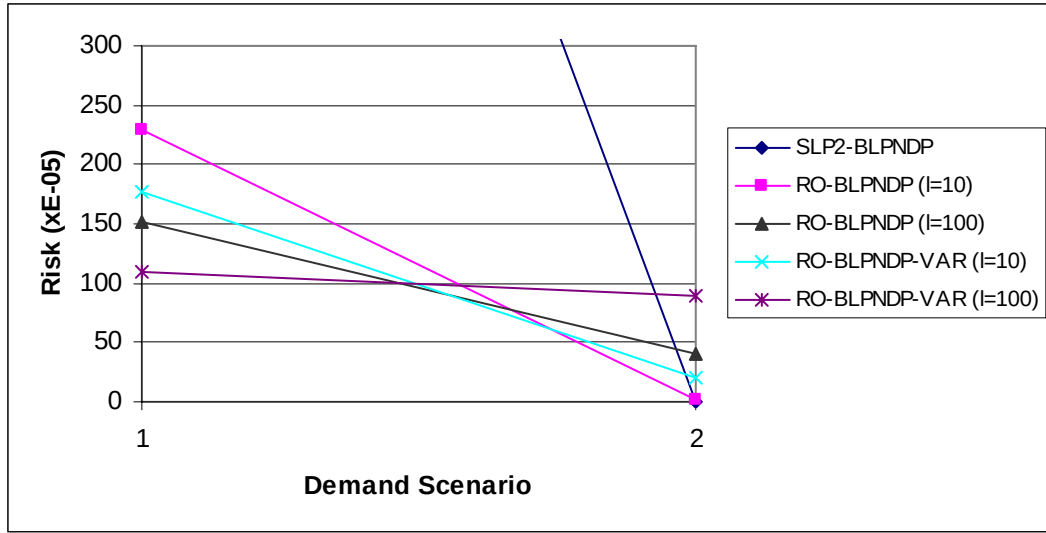


Figure 5.11. Risk of SLP2-BLPNDP, RO-BLPNDP($\lambda=10, 100$) and RO-BLPNDP-VAR($\lambda=10, 100$) Solutions (Budget = 43 and $TSTT^{TARGET}=13.025$)

5.2 Experimental Results of the Integrated NDP, SSD and DTA Models

We consider a test problem, which is a multi-origin single-destination cell transmission network. This simple network allows us to intuitively understand how the network performance will vary when simultaneously designing the network improvement, optimizing the traffic signal settings, and solving dynamic traffic assignment, based on different optimization models. All MIP models are implemented in GAMS/CPLEX and MINLP models in GAMS/DICOPT on a machine with Windows OS, Pentium 4-M CPU, 1.80 GHz and 640 MB of RAM. We first describe a cell transmission test network, followed by the results of all seven NDP-SSD models and the comparison of the sequential NDP and SSD approach against the integrated NDP-SSD approach.

5.2.1 Cell Transmission Test Network (Network 2)

Figure 5.12 shows a CTM test network, composed of 7 cells and 9 cell connectors, and associated data (jam density, saturation flow, travel demand and network-improvement unit costs). This test network contains three intersections ($c3, c5$

and $c6$), two origins (cells $c1$ and $c2$) and single destination (cell $c7$). All cells except the sink cell are considered for capacity expansion. The simulation period is eight time intervals ($|T| = 8$). The pre-determined orderings of signal phases for the three intersections are shown in Figure 5.12f. The assumed boundary values of green times, cycle lengths and offsets for the three intersections are shown in Figure 5.12g. The pre-determined signal timing plans for intersection $c3$ are shown in Figure 5.13, where an oval represents phase 1, and a rectangle represents phase 2. The twelve signal timing plans for intersection $c3$ are labeled $s1, s2, \dots, s12$, respectively. The pre-determined signal timing plans for intersection $c5$ and $c6$ are the same as those for intersection $c3$, and are labeled $s1001, s1002, \dots, s1012$ and $s2001, s2002, \dots, s2012$, respectively.

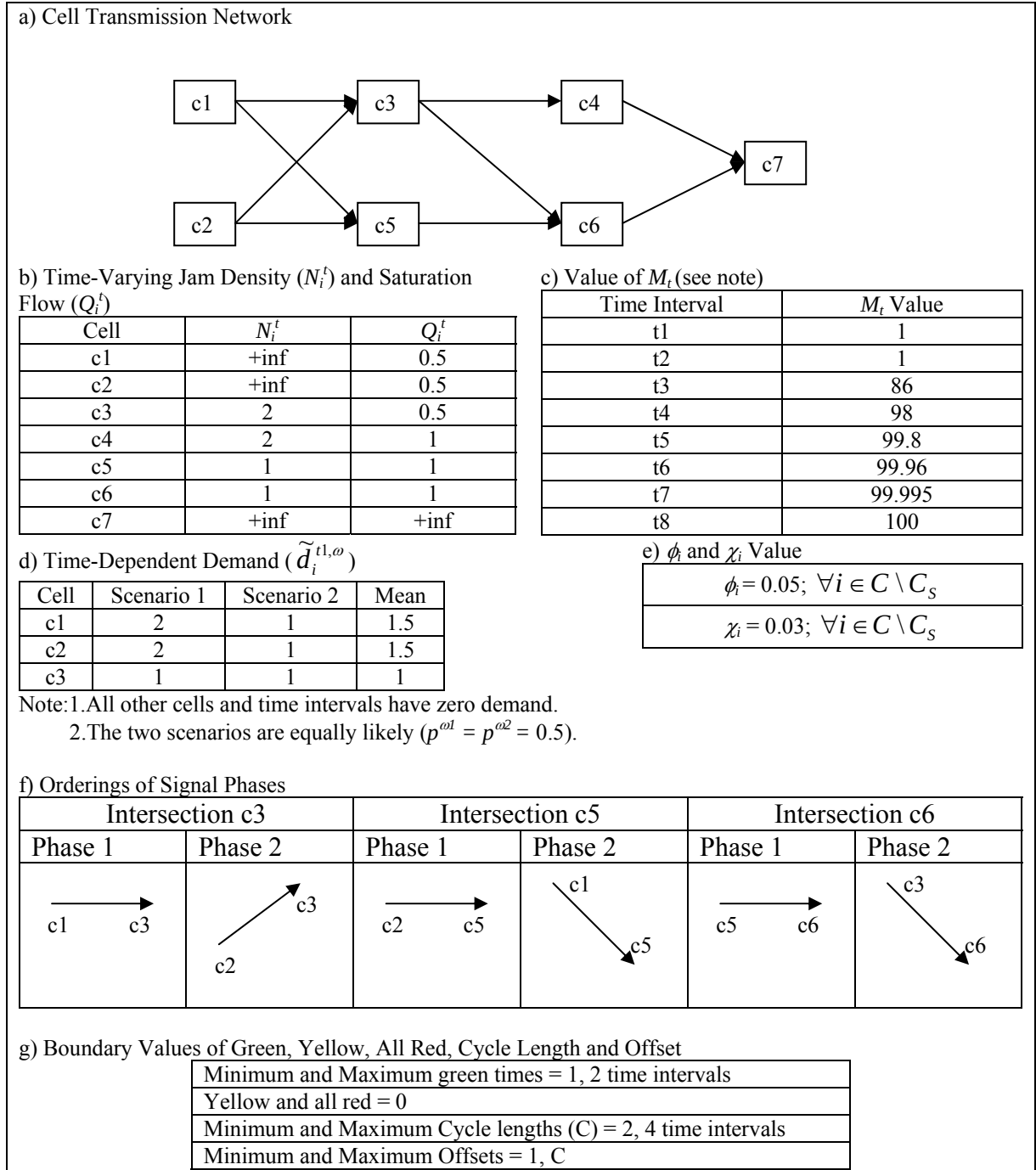


Figure 5.12. Test Network 2

t1	t2	t3	t4	t5	t6	t7	t8	Cycle length (time interval)	Offset of phases 1 and 2 (time interval)
								2	1, 2
								2	2, 1
								3	1, 2
								3	2, 3
								3	3, 1
								3	2, 1
								3	3, 2
								3	1, 3
								4	1, 3
								4	2, 4
								4	3, 1
								4	4, 2

Note: The twelve signal timing plans for intersection c3 are labeled $s_1, s_2, \dots, s_{12}$; those for intersection c5 as $s_{1001}, s_{1002}, \dots, s_{1012}$; and those for intersection c6 as $s_{2001}, s_{2002}, \dots, s_{2012}$.

Figure 5.13. Signal Timing Plans for Intersection c3

5.2.2 Comparison of Seven NDP-SSD Models

The experiment is conducted to compare the proposed six NDP-SSD models and the existing NDP-SSD model. For the deterministic NDP-SSD models, the mean values of time-dependent demands in Figure 5.12d are employed. The TAB range of $[6, 15]$ employed in this section is carefully selected such that different NDP-SSD models recommend different solutions (budget allocation policies and signal settings). With such TAB range, one can see the benefit of the robust optimization model over the other optimization models. If TAB is increased to a high value, all optimization models will recommend solutions that yield the free flow condition. The TAB level that yields the free flow condition is not considered relevant since we only want to show the usefulness of the robust optimization model. With the TAB range of $[6, 15]$, we can achieve this goal.

The seven NDP-SSD models are solved to obtain exact solutions using the solution methods in Section 3.5. The single-level NDP-SSD MIP models (SONDP-SSD, UONDP-SSD, SLP2-SONDP-SSD and SLP2-UONDP-SSD) are coded in GAMS and solved by CPLEX. The MZCL-BLPP of NDP-SSD models (BLPNDP-SSD and SLP2-BLPNDP-SSD) are reduced to equivalent parametric linear bi-level programs, L-BLPP(M), for sufficiently large constants M , which are then transformed to equivalent single-level MIPs that are then solved by GAMS/CPLEX. The MZC-QL-BLPP of robust NDP-SSD model (RO-BLPNDP-SSD) is reduced to an equivalent parametric quadratic-linear bi-level program, QL-BLPP(M), for sufficiently large constants M , which is then transformed to an equivalent single-level MINLP, which contain all linear terms except the convex quadratic objective function. This is solved by GAMS/DICOPT to obtain a global optimum. For RO-BLPNDP-SSD, $TSTT^{TARGET}$ is equal to the optimal expected TSTT from SLP2-BLPNDP-SSD. The λ values between 1 and 1000 are used in RO-BLPNDP-SSD, but the RO-BLPNDP-SSD gives the same solution for all λ values. This may be because for this specific problem instance, the RO-BLPNDP-SSD is insensitive to such λ values. Typically, transportation planners can specify different λ values depending on their preferences. A higher λ value forces the model to select the lower-

risk solution at the expense of the greater expected TSTT. The detailed results are shown in Appendix E.

In the SLP2-SONDP-SSD and SLP2-UONDP-SSD models, the time-varying demands are stochastic; these mean the parameters in the right-hand side of the constraints are random. That is, in the standard form (Equation 2.15) $\tilde{\xi} = \text{vec}(\tilde{d}, \tilde{B})$. As previously discussed, $h(x, \tilde{\xi})$ is convex on the support of $\tilde{\xi} (\Xi_{\tilde{\xi}})$, and Jensen's inequality results in $Eh(x, \tilde{\xi}) \geq h(x, E\tilde{\xi})$. Consequently, the SONDP-SSD and UONDP-SSD models employing the mean values of demands yield underestimated solutions (in terms of their objective functions) to SLP2-SONDP-SSD and SLP2-UONDP-SSD models, respectively. That is, the SONDP-SSD solution yields less TSTT than the expected TSTT from SLP2-SONDP-SSD as shown in Table 5.2. The UONDP-SSD model yields its objective value $(\sum_{(i,j) \in E_S} \sum_{t \in T} (M_t \cdot y_{ij}^t))$ less than $\sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} (p^\omega \cdot M_t \cdot y_{ij}^{t,\omega})$ of SLP2-UONDP-SSD, but the UONDP-SSD solution may not necessarily yield less TSTT than the expected TSTT of SLP2-UONDP-SSD solutions. From our experiment, Table 5.2 shows that the UONDP-SSD does yield underestimated TSTT when compared with $E[TSTT]$ of SLP2-UONDP-SSD over the budget range.

Let us consider the recourse function of SLP2-BLPNDP-SSD, and denote by $hp(b, s, p, \tilde{\xi}^\omega)$ the nested program, which is the same as the recourse function of SLP2-UONDP-SSD. By applying Jensen's inequality to the nested program, $hp(b, s, p, E\tilde{\xi}) \leq E[hp(b, s, p, \tilde{\xi})]$. Note that $hp(b, s, p, E\tilde{\xi}) = \sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t*}$ where

x_i^{t*}, y_{ij}^{t*} are optimal solution to the nested program for the mean value of demands, and

$E[hp(b, s, p, \tilde{\xi})] = \sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} p^\omega M_t y_{ij}^{t,\omega*}$ where $x_i^{t,\omega*}, y_{ij}^{t,\omega*}$ are optimal solution to the

nested program for demand scenario ω . As such, $\sum_{(i,j) \in E_S} \sum_{t \in T} M_t y_{ij}^{t*} \leq$

$\sum_{\omega \in \Omega} \sum_{(i,j) \in E_S} \sum_{t \in T} p^\omega M_t y_{ij}^{t,\omega*}$; this does not necessarily imply that $\sum_{(i,j) \in E_S} \sum_{t \in T} y_{ij}^{t*} \leq$

$\sum_{\omega \in \Omega(i,j) \in E_S} \sum_{t \in T} p^\omega y_{ij}^{t,\omega^*}$. Therefore, the recourse function of SLP2-BLPNDP-SSD may not

be convex on the support of $\tilde{\xi}$ and Jensen's inequality may not be guaranteed. The BLPNDP-SSD model employing the mean values of demands does not necessarily yield underestimated solutions to the SLP2-BLPNDP-SSD. However, the results of our experiment in Table 5.2 show that the BLPNDP-SSD model with the mean values of demands underestimates the TSTT when compared with the expected TSTT from SLP2-BLPNDP-SSD over the budget range between 6 and 15.

Next, for a fair comparison, the solutions of the seven NDP-SSD models over ten total available budget levels are evaluated under the stochastic condition to obtain TSTTs for the two demand scenarios ($TSTT^{\omega 1}$ and $TSTT^{\omega 2}$). Then, $E[UO_TSTT]$, $E[Risk]$ and $E[Robust_Obj(\lambda)]$ are calculated from Equations (5.1)-(5.3). Figures 5.14a-5.14b show the expected UO_TSTT of all NDP-SSD solutions, Figures 5.14c-5.14d depict the expected risks, and Figures 5.14e-5.14f show the expected robust objectives with $\lambda = 10$. For this particular test problem, SLP2-BLPNDP-SSD and SLP2-SONDP-SSD yield the same solutions over the budget range. Since SLP2-SONDP-SSD provides a lower bound for SLP2-BLPNDP-SSD, it implies that for this specific network and budget range, the lower bound solutions (SLP2-SONDP-SSD) are tight and achievable. The SLP2-BLPNDP-SSD and SLP2-SONDP-SSD solutions yield the least expected TSTT, and perform the second best in terms of expected risk. RO-BLPNDP-SSD yields the least expected risk over the budget range at the expense of higher expected TSTTs. The expected TSTTs and expected risks of deterministic NDP-SSD solutions (SONDP-SSD, UONDP-SSD and BLPNDP-SSD) appear fluctuated over the budget range. In terms of robustness evaluation with $\lambda=10$, Figures 5.14e-5.14f show that RO-BLPNDP-SSD yields the most robust solution. This emphasizes the importance of RO-BLPNDP-SSD.

Next, the risks of different NDP-SSD solutions are plotted for each demand scenario for $TAB = 9$ in Figures 5.15a-5.15b. When the slope of the line in Figure 5.15b is steeper, the TSTTs of the two scenarios are more wildly different. The first, second and third worst solutions are UONDP-SSD, SLP2-UONDP-SSD and BLPNDP-SSD. As

expected, RO-BLPNDP-SSD yields the flattest line. Interestingly, SONDP-SSD, which is a deterministic model, yields the second flattest line; SLP2-SONDP-SSD and SLP2-BLPNDP-SSD the third. This implies SLP2-BLPNDP-SSD and SLP2-SONDP-SSD are not always the second best in terms of robustness.

In addition, we show the signal timing plans and budget allocation policies from different NDP-SSD models for $TAB=9$ in Figures 5.15c-5.15d. Obviously, different models recommend various budget allocation policies, and signal timing plans. It can be observed that all models utilize traffic signal coordination by choosing different timing plans with different time offsets for the intersection pairs (c3, c6) and (c5, c6) in order to progress the greatest number of vehicles through the system with the fewest stops and shortest amount of delay. Also, SONDP-SSD and SLP2-BLPNDP-SSD yield solutions with a common cycle length, whereas the other models yield solutions with different cycle lengths. This implies that the commonly used assumption of common cycle length constrains the model too much, and should be relaxed in RO-BLPNDP-SSD to achieve superior robustness.

Table 5.2. TSTT and $E[TSTT]$ of Deterministic and Two-Stage Stochastic NDP-SSD models

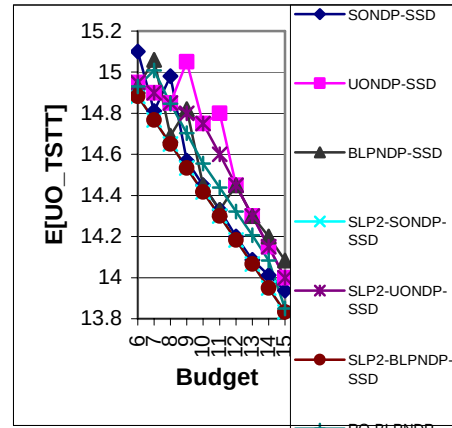
TAB	TSTT of SONDP-SSD	$E[TSTT]$ of SLP2-SONDP-SSD	TSTT of UONDP-SSD	$E[TSTT]$ of SLP2-UONDP-SSD	TSTT of BLPNDP-SSD	$E[TSTT]$ of SLP2-BLPNDP-SSD
6	14.52	14.883	14.7	14.95	14.52	14.883
7	14.34	14.767	14.65	14.9	14.34	14.767
8	14.16	14.65	14.6	14.85	14.16	14.65
9	13.98	14.533	14.55	14.8	13.98	14.533
10	13.8	14.417	14.5	14.75	13.8	14.417
11	13.62	14.3	14.25	14.6	13.62	14.3
12	13.45	14.183	14	14.45	13.45	14.183
13	13.35	14.067	13.75	14.3	13.35	14.067
14	13.3	13.95	13.5	14.15	13.3	13.95
15	13.25	13.833	13.25	14	13.25	13.833

Note: 1. SONDP-SSD, UONDP-SSD and BLPNDP-SSD employ mean values of demands.
2. SLP2-SONDP-SSD, SLP2-UONDP-SSD and SLP2-BLPNDP-SSD employ stochastic demands.
3. TSTT of SONDP-SSD and $E[TSTT]$ of SLP2-SONDP-SSD are under the SODTA condition.
4. TSTT of UONDP-SSD and BLPNDP-SSD, and $E[TSTT]$ of SLP2-UONDP-SSD and SLP2-BLPNDP-SSD are under the UODTA condition.

a) Numerical Values of $E[UO_TSTT]$ of Different NDP-SSD Solutions at Different Budget Levels

TAB	NDP-SSD Models					
	SONDP-SSD	UONDP-SSD	BLPNP-SSD	SLP2-SONDP-SSD; SLP2-BLPNDP-SSD	SLP2-UONDP-SSD	RO-BLPNDP-SSD ($\lambda=1-1000$)
6	15.1	14.95	14.93	14.883	14.95	14.93
7	14.81	14.9	15.06	14.767	14.9	15.009
8	14.98	14.85	14.69	14.65	14.85	14.847
9	14.57	15.05	14.82	14.533	14.8	14.703
10	14.45	14.75	14.45	14.417	14.75	14.556
11	14.33	14.8	14.33	14.3	14.6	14.439
12	14.2	14.45	14.45	14.183	14.45	14.322
13	14.087	14.3	14.3	14.067	14.3	14.206
14	14.012	14.15	14.2	13.95	14.15	14.083
15	13.937	14	14.083	13.833	14	13.85

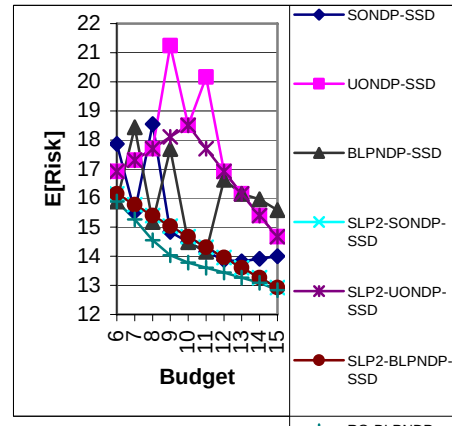
b) Graph of $E[UO_TSTT]$ of Different NDP-SSD Solutions



c) Numerical Values of $E[Risk]$ of Different NDP-SSD Solutions at Different Budget Levels

TAB	NDP-SSD Models					
	SONDP-SSD	UONDP-SSD	BLPNP-SSD	SLP2-SONDP-SSD; SLP2-BLPNDP-SSD	SLP2-UONDP-SSD	RO-BLPNDP-SSD ($\lambda=1-1000$)
6	17.862	16.919	15.888	16.152	16.919	15.888
7	15.529	17.305	18.441	15.772	17.305	15.266
8	18.544	17.701	15.18	15.401	17.701	14.553
9	14.835	21.236	17.683	15.035	18.102	14.037
10	14.488	18.501	14.488	14.669	18.501	13.78
11	14.151	20.161	14.151	14.311	17.701	13.607
12	13.871	16.919	16.629	13.958	16.919	13.436
13	13.823	16.148	16.149	13.605	16.148	13.26
14	13.913	15.401	15.961	13.261	15.401	13.091
15	14.003	14.672	15.589	12.922	14.672	12.837

d) Graph of $E[Risk]$ of Different NDP-SSD Solutions



Note: Target TSTTs are $E[UO_TSTT]$ of SLP2-BLPNDP-SSD in Figure 2a

e) Numerical Values of $E[Robust_Obj(\lambda=10)]$

TAB	NDP-SSD Models					
	SONDP-SSD	UONDP-SSD	BLPNP-SSD	SLP2-SONDP-SSD; SLP2-BLPNDP-SSD	SLP2-UONDP-SSD	RO-BLPNDP-SSD ($\lambda=1-1000$)
6	193.72	184.14	173.81	176.403	184.14	173.81
7	170.1	187.95	199.47	172.487	187.95	167.669
8	200.42	191.86	166.49	168.66	191.86	160.377
9	162.92	227.41	191.65	164.883	195.82	155.073
10	159.33	199.76	159.33	161.107	199.76	152.356
11	155.84	216.41	155.84	157.41	191.61	150.509
12	152.91	183.64	180.74	153.763	183.64	148.682
13	152.317	175.78	175.79	150.117	175.78	146.806
14	153.142	168.16	173.81	146.56	168.16	144.993
15	153.967	160.72	169.973	143.053	160.72	142.22

f) Graph of $E[Robust_Obj(\lambda=10)]$

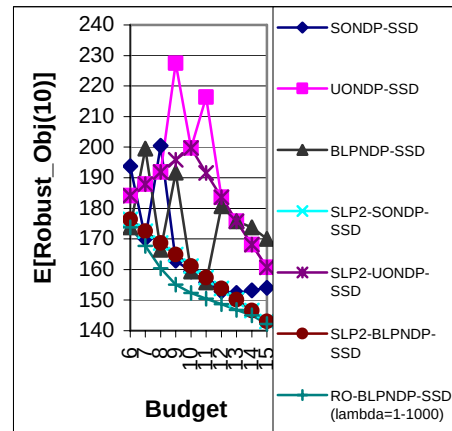


Figure 5.14. $E[UO_TSTT]$, $E[Risk]$ and $E[Robust_Obj(\lambda=10)]$ of Different NDP-SSD Solutions at Different Budget Levels

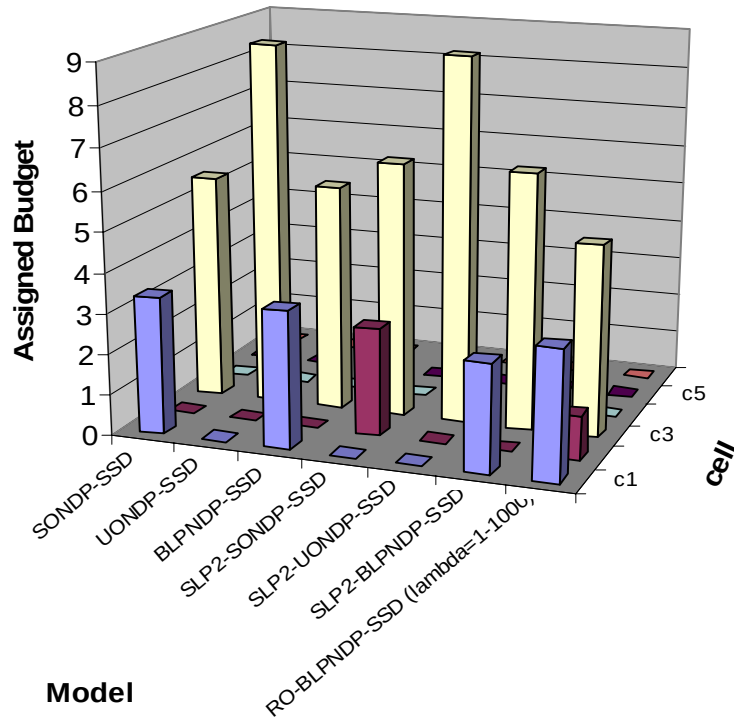
a) Numerical Values of Risks of Different NDP-SSD Solutions for the Two Demand Scenarios

ω	NDP-SSD Models					
	SONDP-SSD	UONDP-SSD	BLPNDP-SSD	SLP2-SONDP-SSD; SLP2-BLPNDP-SSD	SLP2-UONDP-SSD	RO-BLPNDP-SSD ($\lambda=1-1000$)
1	29.6698	42.4713	35.3669	30.0743	36.2043	28.0688
2	0	0	0	0	0	0

c) Signal Timing Plans

NDP-SSD Model	Intersection		
	c3	c5	c6
SONDP-SSD	s8	s1008	s2006
UONDP-SSD	s9	s1008	s2006
BLPNDP-SSD	s8	s1009	s2010
SLP2-SONDP-SSD	s11	s1011	s2006
SLP2-UONDP-SSD	s1	s1001	s2006
SLP2-BLPNDP-SSD	s8	s1008	s2006
RO-BLPNDP-SSD ($\lambda=1-1000$)	s10	s1006	s2008

d) Budget Allocation Policies



b) Graph of Risks of Different NDP-SSD Solutions for the Two Demand Scenarios

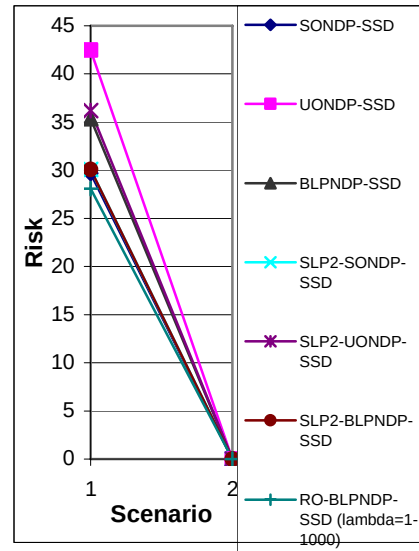


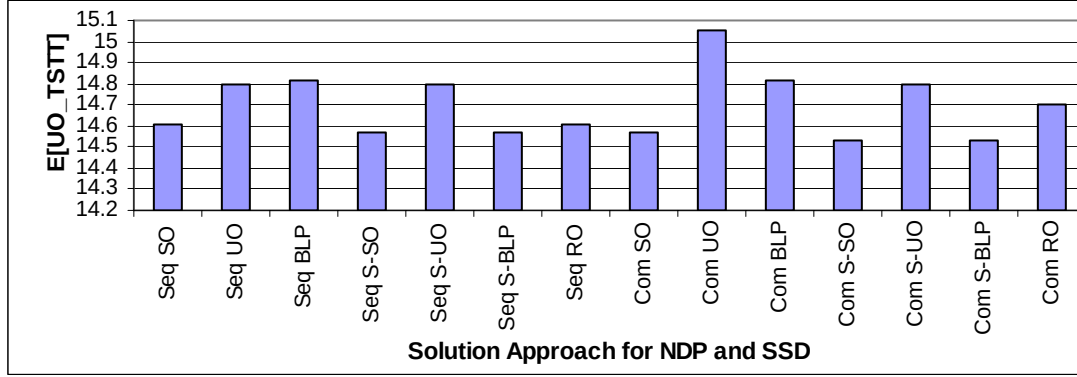
Figure 5.15. Risks of Different NDP-SSD Solutions for the Two Demand Scenarios; Budget Allocation Policies and Signal Timing Plans of Different NDP-SSD Solutions (Total Available Budget = 9 and $TSTT^{TARGET} = 14.533$)

5.2.3 Comparison of the Sequential NDP and SSD Approach against the Integrated NDP-SSD Approach

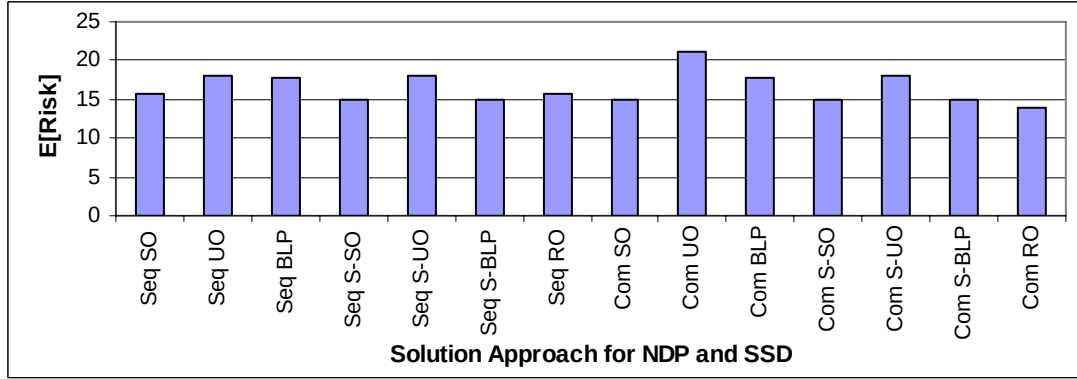
The value of the integrated NDP-SSD models over a sequential approach is illustrated. We are interested in the sequential approach that first solves NDP models and then SSD models. The NDP models include SONDP, UONDP, BLPNDP, SLP2-SONDP, SLP2-UONDP and SLP2-BLPNDP and RO-BLPNDP. These NDP models are simply NDP-SSD models without the SSD components. Similarly, the SSD models include SOSSD, UOSSD, BLPSSD, SLP2-SOSSD, SLP2-UOSSD, SLP2-BLPSSD and RO-BLPSSD, which are basically NDP-SSD models without the NDP components. It is noted that NDP, SSD and NDP-SSD models are all based on DTA models. Specifically, in the sequential approach, we first solve simultaneously NDP and DTA, then SSD and DTA, whereas in the combined approach, NDP, SSD and DTA are solved at once.

We choose $TAB=9$ for this comparison. After solving both sequential and combined approaches, all solutions are evaluated under the stochastic condition to obtain TSTTs for the two demand scenarios. The $E[UO_TSTT]$, $E[Risk]$ and $E[Robust_Obj(\lambda=10)]$ are calculated using Equations 5.1-5.3, and these are shown in Figure 5.16. From Figures 5.17a-5.17b, the robustness can be viewed as the flatness of the lines connecting the risks of the two demand scenarios in Figure 5.17a. In terms of robustness, as expected the RO-BLPNDP-SSD is the best; SONDP-SSD the second; the sequence SLP2-SONDP & SLP2-SOSSD the third; the sequence SLP2-BLPNDP & SLP2-BLPSSD the fourth; and SLP2-BLPNDP-SSD and SLP2-SONDP-SSD the fifth. Interestingly, the sequence RO-BLPNDP & RO-BLPSSD is the sixth. This confirms the fact that the sequential robust NDP and SSD approach yields a suboptimal solution to the integrated robust NDP-SSD approach. Thus, the sequential robust approach (RO-BLPNDP & RO-BLPSSD) may not be desirable, and the combined robust approach (RO-BLPNDP-SSD) should be sought-after. Additionally, the solutions of the sequential NDP and SSD approaches and the integrated NDP-SSD approaches at $TAB = 9$ are depicted in Figure 5.17c.

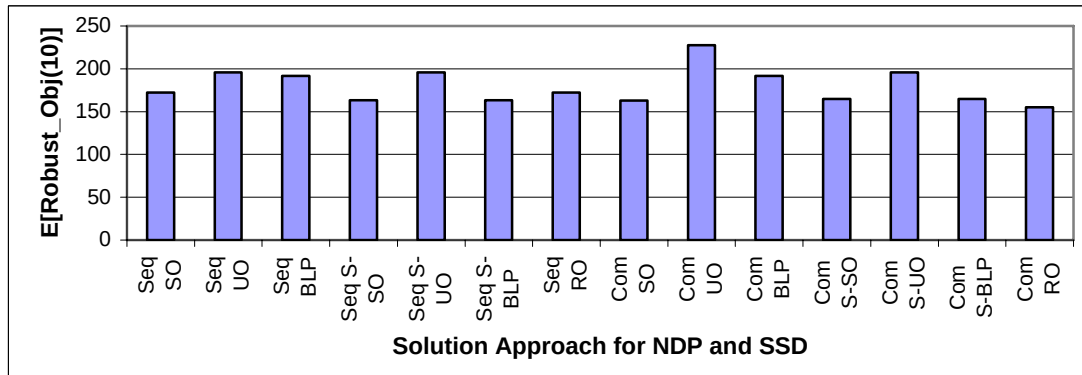
a) Expected Total System Travel Times of Different Sequential and Combined Approaches



b) Expected Risks of Different Sequential and Combined Approaches



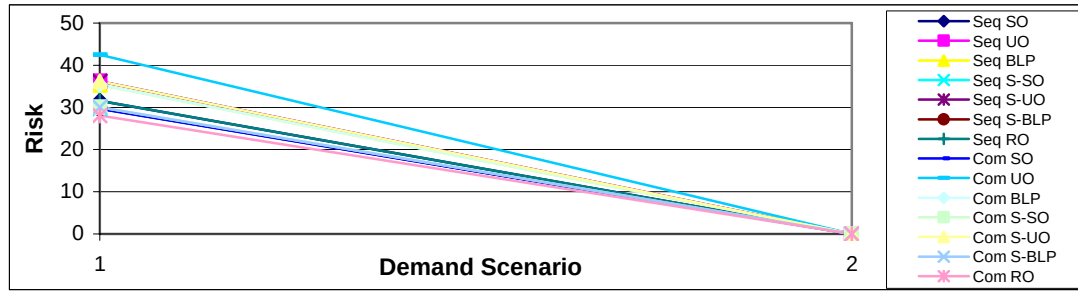
c) $E[\text{Robust_Obj}(\lambda=10)]$ of Different Sequential and Combined Approaches



Note: Seq SO = SONDP & SOSSD; Seq UO = UONDP & UOSSD; Seq BLP = BLPNDP & BLPSSD;
Seq S-SO = SLP2-SONDP & SLP2-SOSSD; Seq S-UO = SLP2-UONDP & SLP2-UOSSD;
Seq S-BLP = SLP2-BLPNDP & SLP2-BLPSSD; Seq RO = RO-BLPNDP & RO-BLPSSD;
Com SO = SONDP-SSD; Com UO = UONDP-SSD; Com BLP = BLPNDP-SSD;
Com S-SO = SLP2-SONDP-SSD; Com S-UO = SLP2-UONDP-SSD;
Com S-BLP = SLP2-BLPNDP-SSD; Com RO = RO-BLPNDP-SSD.

Figure 5.16. Evaluation Results of the Sequential NDP and SSD Approach against the Combined NDP-SSD Approach (Total Available Budget = 9, $TSTT^{TARGET} = 14.533$)

a) Graph of Risks of the Sequential NDP and SSD Approaches and the Integrated NDP-SSD Approaches for the Two Demand Scenarios



b) Numerical Values of Risks of the Sequential NDP and SSD Approaches and the Integrated NDP-SSD Approaches for the Two Demand Scenarios

Solution Approach to NDP and SSD	Demand Scenario	
	1	2
Sequential NDP & SSD Approach		
SONDP & SOSSD	31.5507	0
UONDP & UOSSD	36.2043	0
BLPNDP & BLPSSD	35.3668	0
SLP2-SONDP & SLP2-SOSSD	29.7461	0
SLP2-UONDP & SLP2-UOSSD	36.2043	0
SLP2-BLPNDP & SLP2-BLPSSD	29.7461	0
RO-BLPNDP & RO-BLPSSD ($\lambda=100$)	31.5507	0
Combined NDP-SSD Approach		
SONDP-SSD	29.6698	0
UONDP-SSD	42.4713	0
BLPNDP-SSD	35.3668	0
SLP2-SONDP-SSD	30.0743	0
SLP2-UONDP-SSD	36.2043	0
SLP2-BLPNDP-SSD	30.0743	0
RO-BLPNDP-SSD ($\lambda=100$)	28.0688	0

c) Budget Allocation Policies and Signal Timing Plans of the Sequential NDP and SSD Approaches and the Integrated NDP-SSD Approaches

Solution Approach	Assigned Budget			Intersection		
	b1	b2	b3	c3	c5	c6
Sequential NDP & SSD Approach						
SONDP & SOSSD	1.7	1.7	5.6	s1	s1001	s2006
UONDP & UOSSD	0	0	9	s1	s1001	s2006
BLPNDP & BLPSSD	0	3.4	5.6	s5	s1011	s2006
SLP2-SONDP & SLP2-SOSSD	3.333	0.067	5.6	s8	s1008	s2006
SLP2-UONDP & SLP2-UOSSD	0	0	9	s2	s1002	s2006
SLP2-BLPNDP & SLP2-BLPSSD	0.067	3.333	5.6	s11	s1011	s2006
RO-BLPNDP & RO-BLPSSD ($\lambda=100$)	1.7	1.7	5.6	s12	s1012	s2006
Combined NDP-SSD Approach						
SONDP-SSD	3.4	0	5.6	s8	s1008	s2006
UONDP-SSD	0	0	9	s9	s1008	s2006
BLPNDP-SSD	3.4	0	5.6	s8	s1009	s2010
SLP2-SONDP-SSD	0	2.667	6.333	s11	s1011	s2006
SLP2-UONDP-SSD	0	0	9	s1	s1001	s2006
SLP2-BLPNDP-SSD	2.667	0	6.333	s8	s1008	s2006
RO-BLPNDP-SSD ($\lambda=100$)	3.187	1.062	4.75	s10	s1006	s2008

Note: $b4=b5=0$

Figure 5.17. Evaluation Results of the Sequential NDP and SSD Approach and the Combined NDP-SSD Approach (Total Available Budget = 9, $TSTT^{TARGET} = 14.533$)

5.3 Summary

The experimental results of deterministic, stochastic and robust network design models on the cell transmission test network, Network 1, are presented. For the deterministic bi-level network design model, the modified K^{th} -best algorithm is employed without the guarantee of exact solutions. However, it intuitively provides a possible steepest-descent-type iterative heuristic for multi-destination networks. Since this involves the cumbersome pre-processing task on a simplex tableau and it yields at best a local optimum, we do not pursue on this possible heuristic, and aim at developing metaheuristics as discussed in Chapter 4. The solutions from the modified K^{th} best algorithm are confirmed globally optimal by the second solution method, the KKT-based MIP reformulation.

Subsequently, we compared the three new NDP models (BLPNDP, SLP2-BLPNDP and RO-BLPNDP) with the existing four NDP models (SONDP, UONDP, SLP2-SONDP and SLP2-UONDP) on Network 1. With this complexity, we are limited to solving a small network enabling us to gain insight into the problem and to intuitively understand the effects of alternative improvement policies. It is expected that this work will facilitate the progressive development towards useful practical methodologies. Although the findings from our computational experience cannot necessarily be generalized, they provide insightful and interesting information. We find that BLPNDP yields underestimated solutions (in terms of total system travel time) to the SLP2-BLPNDP in our test problem although this is not always guaranteed. When evaluated all NDP solutions under stochastic condition, SLP2-UONDP and UONDP perform the worst; SONDP and BLPNDP the second worst; SLP2-SONDP the third worst. As expected, SLP2-BLPNDP always yields the least expected TSTT. When comparing between SLP2-BLPNDP, RO-BLPNDP($\alpha=10$) and RO-BLPNDP($\alpha=100$), RO-BLPNDP($\alpha=100$) yields the least expected risk as it places higher weight on the risk term in the objective function, and the SLP2-BLPNDP yields the highest expected risk. In terms of robustness, we show that RO-BLPNDP yields the most robust solution, and no other model provides a satisfactory solution across the budget range. We also

illustrate how a visualized graph may be used to elicit the preference information from transportation planners on the desired degree of robustness.

Additionally, we consider the alternative robust NDP model (RO-BLPNDP-VAR) that optimizes the tradeoff between the expected value and variance of TSTT. RO-BLPNDP-VAR is compared with RO-BLPNDP and SLP2-BLPNDP on Network 1. RO-BLPNDP-VAR yields the least variance of TSTT as it is a component in the objective function. We suggest that RO-BLPNDP-VAR and RO-BLPNDP are appropriate on different settings that highly depend on the preference of transportation planners.

Then, we compare the integrated NDP, SSD and DTA models among themselves and against the sequential NDP and SSD approach on the simple cell transmission network, Network 2. With this complexity, we are limited to solving a small network so as to gain insight into the problem and to intuitively understand the effects of alternative improvement policies and signal settings. We find that BLPNDP-SSD yields underestimated solutions (in terms of total system travel time) to the SLP2-BLPNDP-SSD in our test problem although this is not always guaranteed. From the comparison among the seven integrated NDP-SSD models, RO-BLPNDP-SSD yields the most robust solutions over the budget range, and SLP2-BLPNDP-SSD and SLP2-SONDP-SSD do not always yield the second most robust solution. In SSD, all models utilize the traffic signal coordination for traffic progression, and the assumption of common cycle length should be relaxed as it constrains the model too severely. From the comparison of the sequential NDP and SSD approach against the combined NDP-SSD approach, the integrated robust model (RO-BLPNDP-SSD) performs the best in terms of robustness, and interestingly, the sequential robust approach (RO-BLPNDP & RO-BLPSSD) yields the fifth most robust solution. This reiterates that the sequential robust approach may not be desirable, and the integrated robust approach should be sought after whenever possible.

Indeed, the analytical models cannot handle the real-size network and is limited to a single destination problem. However, it helps facilitate the development of a practical heuristic based on math programming, and/or metaheuristics. Also, it can provide a

benchmark for a comparison of future heuristics. The next chapter documents the experimental results of metaheuristics developed in this dissertation.

Chapter 6. Experimental Results of Metaheuristic Approach

In Chapter 4, the five metaheuristic algorithms are developed, namely, simulated annealing (SA), modified simulated annealing (MSA), genetic algorithm (GA), reactive tabu search (RTS) and random search (RS). For RTS, we proposed three neighborhood definitions (move types 0, 1 and 2), resulting in three variations of RTS: RTS-MT0, RTS-MT1 and RTS-MT2. In this chapter, the comparison of the seven metaheuristic algorithms is shown: SA, MSA, GA, RTS-MT0, RTS-MT1, RTS-MT2 and RS. The test networks for the study of metaheuristic approach are first described. Then, the implementation and calibration of SA, MSA and GA parameters are discussed, followed by the results of the preliminary metaheuristic performance comparison for the deterministic NDP. Next, we thoroughly compare the performance of different metaheuristics for deterministic, stochastic and robust problems of SSD, NDP and NDP-SSD on two limited multi-origin multi-destination networks in order to identify the best metaheuristic algorithm for each problem on each network. Finally, we employ the identified best metaheuristics to find the near-global optimal solutions for all problems on the two networks, and evaluate these solutions in terms of $E[Robust_Obj]$, $E[TSTT]$, $E[Risk]$ and $V[TSTT]$.

6.1. Test Networks

Five test networks are considered. The first is a multi-origin single-destination cell transmission network, Network 1, in Figure 5.1 with deterministic demands in Figure 5.2 in Chapter 5. The other four networks are multi-origin multi-destination limited networks: two hypothetical grid networks (Networks 3.1 and 3.2) and two larger-size, real networks (modified Sioux Fall networks, Networks 4.1 and 4.2).

Network 3.1

Figure 6.1a shows Network 3.1 composed of 6 nodes and 14 links. All links are candidates for capacity expansion. All links are 1-lane and 2-mile long with the free flow

speed of 49.5 mph. All horizontal links have the capacity of 700 vehicles per hour (vph); and all vertical links 600 vph. The simulation period is a peak hour with 12 O-D pairs (1-2, 1-5, 1-6, 2-1, 2-5, 2-6, 5-1, 5-2, 5-6, 6-1, 6-2 and 6-5). Each O-D demand is 450 vehicle trips per hour, yielding the total 5400 vehicle trips. The static demands are distributed over the 12 five-minute time slices (i.e. total one hour) by the following weights: 0.05, 0.05, 0.05, 0.1, 0.1, 0.15, 0.15, 0.1, 0.1, 0.05, 0.05 and 0.05, respectively. Within each time slice, the demands are assumed uniformly distributed. Network 3.1 is only employed in the preliminary study on the deterministic NDP in Section 6.3.2 with the parameters: $TAB = 1500$; χ_i = increase in N_i^t per one unit of b_i ($=\chi$ in this study) = 40; and ϕ_i = increase in Q_i^t per one unit of b_i ($=\phi$ in this study) = 10. The total free flow travel time (TFFT) is calculated by assuming all vehicles travel at the free flow speed, yielding the TFFT of 436.36 hours.

Network 3.2

Figure 6.1b shows Network 3.2 composed of 9 nodes and 24 links. All links are candidates for capacity expansion, and nodes 2, 4 and 5 are signalized intersections. All links are 2-lane and 2 mile long with the free flow speed of 49.5 mph (0.825 mile/min). All horizontal links have the capacity of 1400 vph; and all vertical links 1200 vph. The simulation period is a peak hour. We consider two sets of O-D matrices: 1) 72 O-D pairs (all nodes to all other nodes) for deterministic problems, and 2) 12 O-D pairs (1-3, 1-7, 1-9, 3-1, 3-7, 3-9, 7-1, 7-3, 7-9, 9-1, 9-3 and 9-7). The first set of O-D matrix (72 O-D pairs) is only employed in the metaheuristic performance comparison for the deterministic SSD, NDP and NDP-SSD problems in Section 6.4. For stochastic and robust problems, each O-D demand is a random variable, which is normally distributed with the mean of 450 vph and the standard deviation of 50 vph. The static demands are distributed over the peak hour in the same way as Network 3.1.

For Network 3.2, we employ the NDP parameters: $TAB = 1500$, $\chi = 160$, and $\phi = 160$. Note that only in the metaheuristic performance comparison for the deterministic

SSD, NDP and NDP-SSD problems in Section 6.4, we employ $\chi = 40$ and $\phi = 40$. The below SSD parameters are used for Network 3.2:

- all red time = 2 seconds per movement phase,
- yellow time = 4 seconds per movement phase,
- Node 5 (a 4-leg intersection)
 - Min green times for movement phases 1, 2, 3, 4 = 10, 15, 10, 15 seconds
 - Min and Max cycle length = 74 and 250 seconds
- Nodes 2 and 4 (3-leg intersections)
 - Min green times for movement phases 1, 2, 3 = 15, 10, 10 seconds
 - Min and Max cycle length = 53 and 190 seconds

The total free flow travel time (TFFT) for Network 3.2 with 12 O-D pairs is calculated by assuming all vehicles travel at the free flow speed. For the deterministic case, the number of total vehicle trips is 5400, and the TFFT is 581.82 hours. For the stochastic and robust cases, we calculate the expected number of total vehicle trips (5,406.63 vehicle trips) and the expected TFFT (585.43 hours) using the 30 O-D demand realizations generated for solution evaluation in Table F.4 in Appendix F.

Network 4.1, Modified Sioux Falls Network

Figure 6.1c shows Network 4.1 composed of 24 nodes and 76 arcs. Ten links with numerical labels are candidates for capacity expansion. The network is the aggregated network of the city of Sioux Falls, South Dakota, used by many researchers in the literature. Since we require information unavailable in the literature (i.e. number of lanes, link length, free flow speed, capacity), these are assumed as follows. All links are 1-lane with the capacity of 600 vph and the free flow speed of 49.5 mph. The link lengths can be determined from the scale in Figure 6.1c. The study period is a peak hour with 528 O-D pairs. All O-D demands are determined from 450 vph multiplied by the factors in Table F.1 in Appendix F, yielding the total 178,542 vehicle trips. The static demands are distributed over the peak hour in the same way as Network 3.1. Network 4.1 is only employed in the preliminary study on the deterministic NDP in Section 6.3.2

with the parameters: $TAB = 1500$, $\chi = 40$ and $\phi = 10$. The total free flow travel time (TFFT) is calculated by assuming all vehicles travel at the free flow speed, yielding the TFFT of 10,805.05 hours.

Network 4.2, Modified Sioux Falls Network

Network 4.2 is the same as Network 4.1 in Figure 6.1c except that all links are 2-lane with the capacity of 1200 vph and the free flow speed of 49.5 mph, and nodes 6, 8, 9, 10, 16, 17 and 18 are signalized intersections. The study period is a peak hour with 33 O-D pairs. All O-D demands are determined from a normal random variable (the mean of 900 vph and standard deviation of 100) multiplied by the factors in Table F.2 in Appendix F. We employ the NDP parameters: $TAB = 1500$, $\chi = 160$ and $\phi = 160$. For SSD parameters,

- all red time = 2 seconds per movement phase,
- yellow time = 4 seconds per movement phase,
- Node 10 (a 5-leg intersection)
 - Min green times for movement phases 1, 2, 3, 4 and 5 = 10 seconds
 - Min and Max cycle length = 80 and 250 seconds
- Nodes 8 and 16 (4-leg intersections)
 - Min green times for movement phases 1, 2, 3, 4 = 10 seconds
 - Min and Max cycle length = 64 and 210 seconds
- Nodes 6, 9, 17 and 18 (3-leg intersections)
 - Min green times for movement phases 1, 2, 3 = 10 seconds
 - Min and Max cycle length = 48 and 180 seconds

The total free flow travel time (TFFT) is calculated by assuming all vehicles travel at the free flow speed. For the deterministic case, the number of total vehicle trips is 22,374, and the TFFT is 1,515.21 hours. For the stochastic and robust cases, we calculate the expected number of total vehicle trips (22,341.53 vehicle trips) and the expected TFFT (1,514.04 hours) using the 30 O-D demand realizations generated for solution evaluation in Table F.6 in Appendix F.

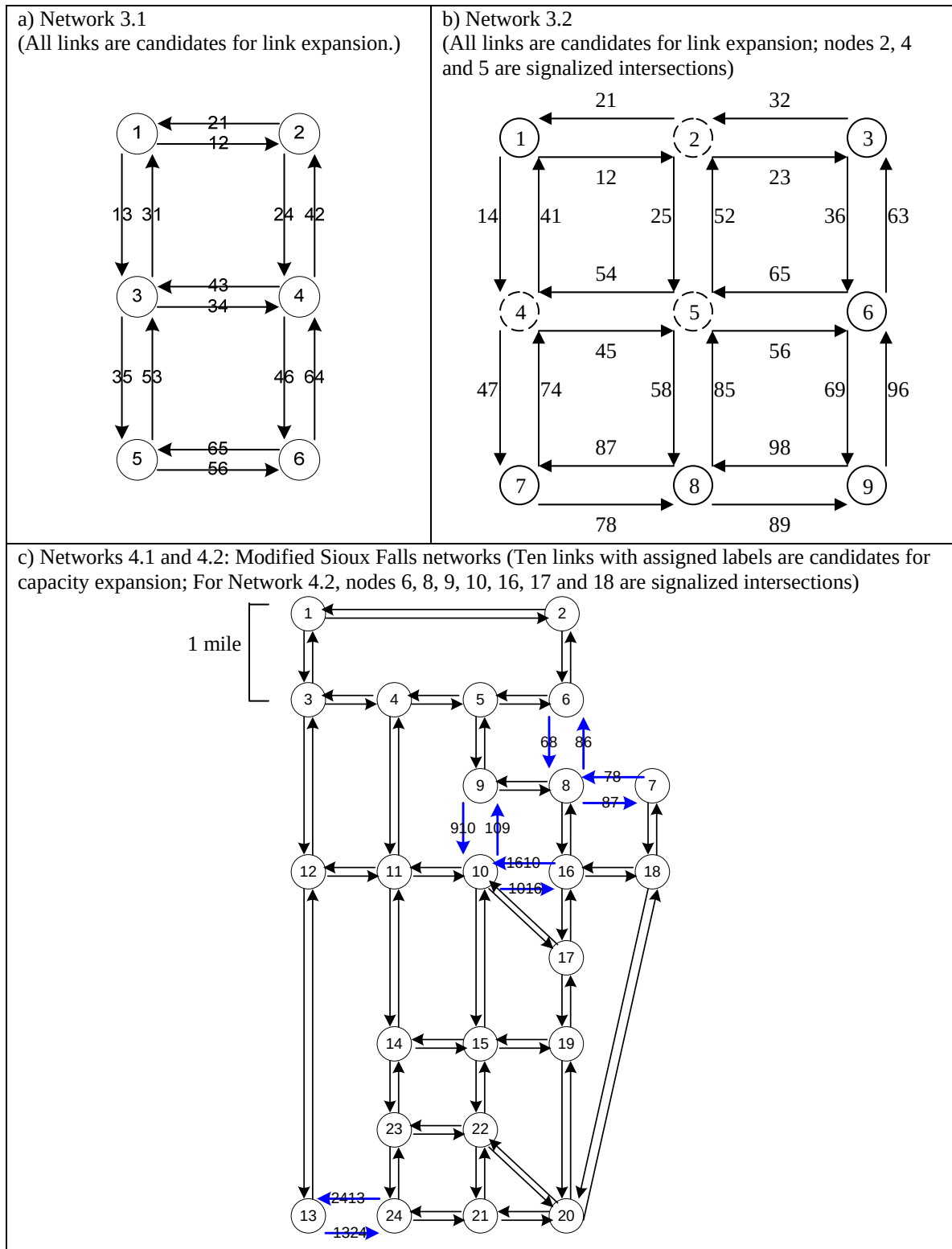


Figure 6.1. Multi-Origin Multi-Destination Test Networks

6.2. Implementation and Calibration of SA, MSA and GA Parameters

Implementation

SA, MSA and RS are implemented in C++, whereas GA and RTS are implemented in C. These run on a Dell PowerEdge 6600 Server with dual 2.3 GHz Xeon processors and 4 GB of RAM, running under Redhat Fedora Core 3. Next, we discuss the calibration process.

Calibration of SA, MSA and GA Parameters

The calibration of SA, MSA and GA parameters are performed for the deterministic NDP on Network 1 since the exact solutions are known from the analytical BLPNDP model in Chapter 5. The calibration process is progressive. First, a set of important parameters is identified. Then, based on certain recommendations in the literature, a base case of parameters is established. With all parameters at base values, the first parameter is arbitrarily selected, then its value is set at different values, and corresponding metaheuristic runs are performed. We record the first parameter value, which yields the best objective function, as the chosen value. Subsequently, we set the first parameter at the chosen value and all other parameters at the base values; then, the second parameter is arbitrarily selected. The second parameter is set to various values, and metaheuristic runs are performed. The second parameter value with the best objective function is recorded as the chosen parameter value. This process is repeated until obtaining the chosen values of all identified parameters. Figures 6.2a, 6.2b and 6.2c show the identified parameters of the respective SA, MSA and GA, their candidate values and their base-case values. The result of the progressive calibration based on Network 1 is the set of chosen parameters that are employed to solve the BLPNDP on Network 1 and may be applied to the other test problems. It is noted that RS and RTS have only one parameter (i.e. total trials), so there is no need for calibration.

a) Calibration of SA Parameters on Network 1 (Budget = 37; $N_{OUT} = 20$)

SA Parameters (considered parameter values)	Base Case	Chosen Parameters
1.Type of Q (Diagonal matrix with the same elements, Lower triangular matrix with the same element)	Diagonal	Lower Triangular
2.Value of elements of the initial Q (1, 0.1, 0.01, 0.001, 0.0001)	0.001	0.1
3.Delta (boundary intensifying) (0.1, 0.01, 0.001)	0.01	0.01
4.Base for exponential function (cooling schedule) (1.5, 2.0, 4.0, 10.0)	2.0	2.0
5. Multiple of N_{IP} (for N_{IN}) (15, 25, 35, 45, 55)	15	15
6.Initial solution strategy (SONDP solution, Uniformly assigned budget)	Uniformly assigned budget	Uniformly assigned budget

b) Calibration of MSA Parameters on Network 1 (Budget = 37; $N_{OUT} = 20$)

Modified SA Parameters (considered parameter values)	Base Case	Chosen Parameters
1.Type of Q (Diagonal matrix with the same elements, Lower triangular matrix with the same element)	Diagonal	Lower Triangular
2.Value of elements of the initial Q (1, 0.1, 0.01, 0.001, 0.0001)	0.001	0.01
3.Base for exponential function (cooling schedule) (1.5, 2.0, 4.0, 10.0)	2.0	2.0
4. Multiple of N_{IP} (for N_{IN}) (15, 25, 35, 45, 55)	15	15
5.Initial solution strategy (SONDP solution, Uniformly assigned budget)	Uniformly assigned budget	Uniformly assigned budget

c) Calibration of GA Parameters on Network 1 (Budget = 37; Total Trials = 900; Number of Genes = 4; Structure Length = 56; Required Precision = 4; Required bits $m = 14$)

GA Parameter (considered parameter values)	Base Case Value	Chosen Parameters
1.Selection Algorithm: (SUS Proportion + Elitism, SUS Proportion w/o Elitism, SUS Ranking + Elitism, and SUS Ranking w/o Elitism)	SUS Proportion with Elitism	SUS Proportion w/o Elitism
2.Population Size (30, 50, 70, 100)	50	50
3.Crossover Rate (0.1, 0.3, 0.5, 0.6, 0.7, 0.9)	0.6	0.6
4.Mutation Rate (0.01, 0.001, 0.0001, 0.00001)	0.001	0.001
5.Initial population (one SONDP solution and the others randomly generated, all randomly generated)	All randomly generated	All randomly generated

Figure 6.2. Calibration of SA, MSA and GA Parameters

In the next section, the performances of different metaheuristics for the deterministic NDP are compared.

6.3 Preliminary Performance Comparisons of Metaheuristics for Deterministic NDP on Networks 1, 3.1 and 4.1

In our preliminary study of metaheuristics, we focus on the deterministic NDP for simplicity. Based on the calibrated SA, MSA and GA parameters in the previous section, RS, SA, MSA, GA, RTS-MT0, RTS-MT1 and RTS-MT2 are tested on Network 1 with the deterministic demands in Figure 5.2, and SA, GA and RS are tested on Networks 3.1 and 4.1.

All metaheuristics share the same functional evaluation that is considered computationally intensive. An iteration of RS, SA, MSA, GA, RTS-MT0, RTS-MT1 and RTS-MT2 contains different number of functional evaluations (trials), so for a fair comparison, all metaheuristic algorithms are run with the same stopping criterion; i.e., the number of trials. For Network 1, UO DTA is an LP and solved by CPLEX. For the other networks, UO DTA is solved by simulation. The DTA module in VISTA can be run by starting from previous solution (employing the generated path set) that is faster than by starting from scratch. However, the equilibration inherently involves randomness, so different UO DTA runs starting from previous solution will yield different TSTTs. Essentially, all analyses in this study employ the UO DTA starting from scratch with the same random seed in order to have comparable results at the expense of longer CPU time than the UODTA starting from previous solution. The performance comparison is made according to the three criteria: solution quality, convergence speed and CPU time.

6.3.1 Preliminary Performance Comparison of Metaheuristics for Deterministic NDP on Network 1

For Network 1, we employ the calibrated parameter sets, and run all metaheuristic algorithms over the budget range [37, 46] with the total trials of 2700, yielding the results in Table 6.1. The optimality gaps can be calculated using the exact solutions in Section 5.1.2 in Chapter 5. Then, the ranking of all seven metaheuristics with respect to the solution quality is performed as follows. At each *TAB*, the algorithm that yields the smallest optimality gap is ranked 1 as best, and the greatest optimality gap ranked 7 as

worst. Then, for each algorithm, the numbers of rankings are counted over the budget range, as shown in Table 6.2. We assign the weight from 3 to -3 to the rankings from 1 to 7. As such, the score of the algorithms is the weighted summation of the number of rankings. The algorithms ranked from the best to the worst in terms of solution quality are RTS-MT1, GA, RTS-MT2, RTS-MT0, MSA, SA and RS.

Subsequently, for a fair comparison in terms of convergence speed and CPU time, we consider only the *TAB* of 37 that was used in SA, MSA and GA parameter calibrations. Table 6.3 and Figure 6.3 show the results of the seven metaheuristics on Network 1 with the *TAB* of 37. In terms of CPU time, the algorithms in the order from the fastest to the slowest are RTS-MT1, RTS-MT2, MSA, RTS-MT0, GA, SA and RS. In terms of convergence speed, the algorithms in the order from the fastest to the slowest are RTS-MT1, GA, RTS-MT2, RTS-MT0, SA, RS and MSA.

When considering all three criteria (solution quality, convergence speed and CPU time), the results on Network 1 suggest that RTS-MT1 outperforms the other six metaheuristic algorithms on the three criteria, and RTS-MT2 and GA are the second best. RTS-MT0 is the fourth best in each of the three criteria. RS, SA and MSA perform very poorly.

Table 6.1. Total System Travel Time of Best Solutions from Metaheuristics, Associated Optimality Gaps, and Performance Ranking Based on Solution Quality on Network 1 over Different Total Budgets

TAB	TSTT (% Optimality Gap) Ranking *						
	SA	MSA	GA	RS	RTS-MT0	RTS-MT1	RTS-MT2
37	22.111 (0.1041) 3	22.138 (0.2264) 5	22.093 (0.02264) 2	22.194 (0.4799) 7	22.146 (0.2626) 6	22.090 (0.0091) 1	22.114 (0.1177) 4
38	22.045 (0.3185) 4	22.102 (0.5779) 6	21.994 (0.08646) 2	22.107 (0.6007) 7	22.0687 (0.4264) 5	21.979 (0.0182) 1	22.005 (0.1365) 3
39	22.029 (0.7593) 6	21.977 (0.5214) 4	21.881 (0.08233) 2	22.053 (0.8690) 7	21.993 (0.5923) 5	21.865 (0.0091) 1	21.900 (0.1689) 3
40	21.806 (0.2575) 3	21.827 (0.3540) 5	21.773 (0.1057) 1	21.926 (0.8092) 7	21.924 (0.8000) 6	21.779 (0.1333) 2	21.809 (0.2713) 4
41	21.902 (1.220) 7	21.751 (0.5222) 4	21.646 (0.03697) 2	21.799 (0.7441) 5	21.872 (1.081) 6	21.639 (0.0046) 1	21.696 (0.2680) 3
42	21.808 (1.315) 7	21.570 (0.2091) 4	21.537 (0.05575) 1	21.629 (0.4832) 6	21.550 (0.1161) 2	21.576 (0.2369) 5	21.556 (0.1440) 3
43	21.550 (0.6398) 7	21.697 (1.3263) 5	21.422 (0.04203) 1	21.508 (0.4437) 6	21.473 (0.2802) 4	21.438 (0.1168) 3	21.427 (0.0654) 2
44	21.452 (0.7136) 7	21.403 (0.4836) 5	21.321 (0.09859) 3	21.443 (0.6714) 6	21.304 (0.0188) 1	21.308 (0.0376) 2	21.402 (0.4789) 4
45	21.548 (1.699) 7	21.518 (1.5575) 6	21.217 (0.1369) 3	21.385 (0.9298) 5	21.213 (0.1180) 2	21.199 (0.0519) 1	21.251 (0.2973) 4
46	21.199 (0.5884) 6	21.135 (0.2847) 4	21.180 (0.4982) 5	21.327 (1.196) 7	21.120 (0.2135) 3	21.076 (0.0047) 1	21.100 (0.1186) 2

Note: Ranking (1=Best; 7= Worst)

Table 6.2. Ranking Results Based on Solution Quality on Network 1 over Different Total Budgets for All Metaheuristics

Ranking	Weight	Number of Rankings over TAB = [37, 46]						
		SA	MSA	GA	RS	RTS-MT0	RTS-MT1	RTS-MT2
1=Best	3	0	0	3	0	1	6	0
2	2	0	0	4	0	2	2	2
3	1	2	0	2	0	1	1	4
4	0	1	4	0	0	1	0	4
5	-1	0	4	1	2	2	1	0
6	-2	2	2	0	3	3	0	0
7=Worst	-3	5	0	0	5	0	0	0
Score (weighted sum)		-17	-8	18	-23	0	22	8
Ranking		6	5	2	7	4	1	3

Table 6.3. Best Solutions from RS, SA, MSA, GA, RTS-MT0, RTS-MT1 and RTS-MT2 on Network 1 (Budget = 37, Total Trials = 2700)

Cell	Metaheuristic Algorithms						
	SA	MSA	GA	RS	RTS-MT0	RTS-MT1	RTS-MT2
c1	10.0375	10.4317	10.0492	10.2448	10.5722	9.9873	9.9735
c2	16.4705	16.1588	16.3392	15.8470	15.8569	16.3801	16.2332
c3	0	0.0203	0	0.4305	0	0	0.0010
c4	0	0.0203	0	0.4305	0	0	0.0010
c5	10.4920	10.3689	10.6116	10.0472	10.5709	10.6326	10.7913
CPU Time (min)	1.456	1.090	1.355	2.691	1.178	1.026	1.035
Best TSTT	22.111	22.138	22.093	22.194	22.146	22.090	22.114
Opt. Gap (%)	0.1041	0.2264	0.02264	0.4799	0.2626	0.0091	0.1177
Trial Found Best	392	2563	2232	773	1457	1681	1401

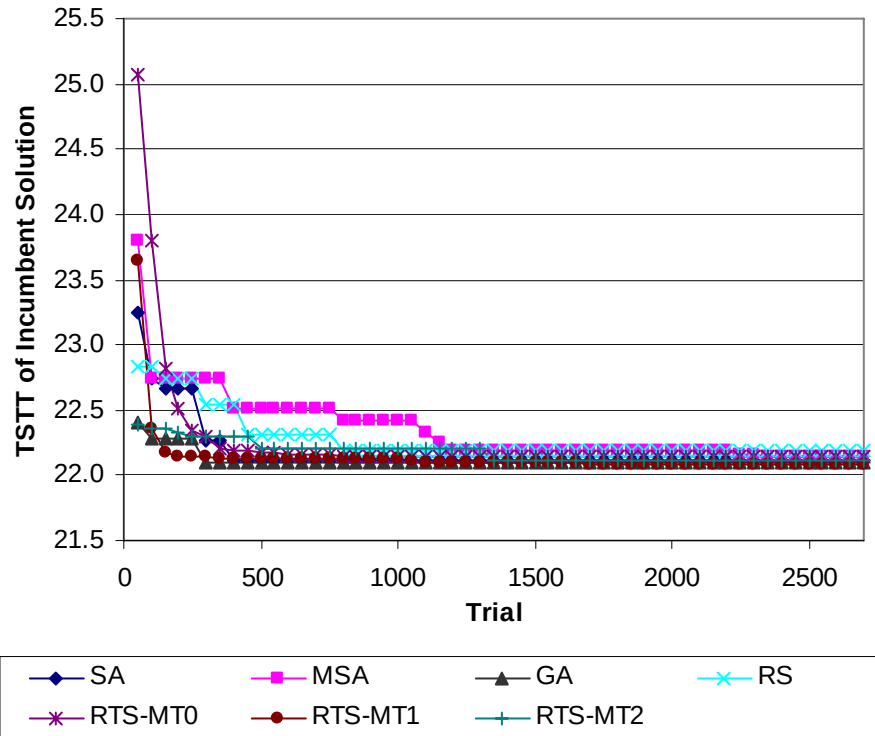
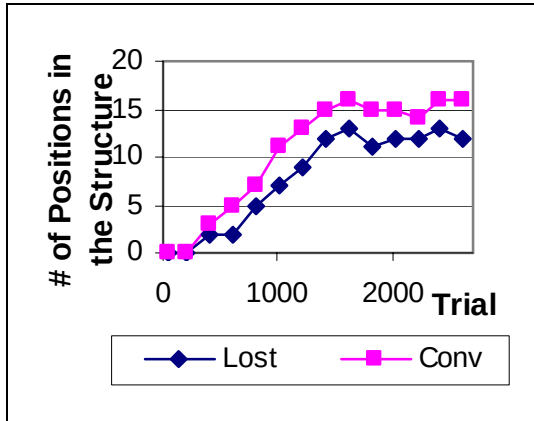


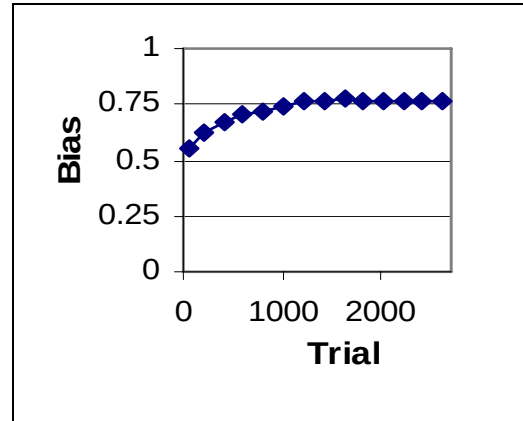
Figure 6.3. Convergence Characteristics of Metaheuristics for Deterministic NDP on Network 1 (Total Budget = 37)

In addition, the diverse convergence characteristics of GA (Grefenstette, 1990) (see the definitions in Step 3 of GA in Section 4.6 in Chapter 4) can be observed on Network 1 with the *TAB* of 37 in Figure 6.4. *Lost* and *Conv* do not converge to certain values, and are equal to respective 12 and 16 in the last generation. *Bias* converges to 0.77. These indicate the population still contains a variety of structures (i.e., not dominated by certain super structures). *Online*, *Offline*, *Best* and *Average* show that GA converges in terms of the objective function, and *Best* appears to be the better measure, also indicating that GA achieves its best solution at trial 2232.

a) *Lost* and *Conv*



b) *Bias*



c) *Online*, *Offline*, *Best* and *Average*

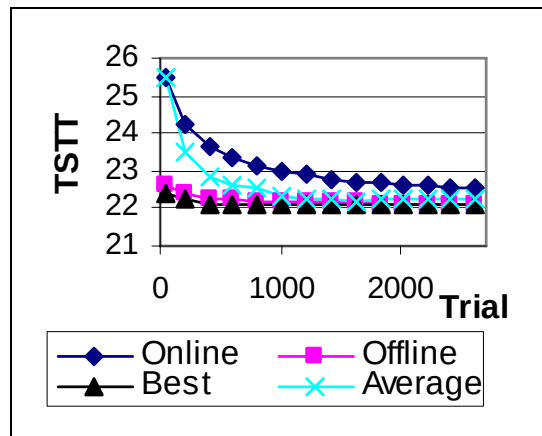


Figure 6.4. Diverse Convergence Characteristics of Genetic Algorithm for Deterministic NDP on Network 1

Next, we assume that the calibrated parameters based on Network 1 are also good for the other networks (this is a typical assumption found in the literature), and only SA, GA and RS are tested for the deterministic NDP on Networks 3.1 and 4.1.

6.3.2 Preliminary Performance Comparisons of SA, GA, and RS for Deterministic NDP on Networks 3.1 and 4.1

For Network 3.1 with the *TAB* of 1500, SA, GA and RS are applied with the total trials of 1000, and the results are shown in Figures 6.5a and 6.5c. GA takes 28.85 hours to discover its best solution at trial 523 (TSTT=1028.2 with 8.645% improvement from the initial state and TSTT/TFFT=2.36). SA takes about 1.5 times longer than GA to find its best solution at trial 848 (TSTT=1079.84 with 4.057% improvement and TSTT/TFFT=2.47). RS takes 1.21 times longer than GA to find its best solution at trial 53 (TSTT=1107.28 with 1.619% improvement and TSTT/TFFT=2.54). Thus, GA performs best among the three algorithms on Network 3.1 in all three criteria, whereas SA performs better than RS in terms of solution quality and convergence speed.

For Network 4.1 with the *TAB* of 1500, SA, GA and RS with the calibrated parameters based on Network 1 are applied for 500 total trials. Here, we rename GA as GA1. From Figures 6.5b and 6.5d, GA1 takes shortest CPU time with fastest convergence speed to find its best solution at trial 448 (TSTT=81353.14 with 0.8236% improvement and TSTT/TFFT=7.5292). SA takes the longest CPU time to find its best solution at trial 540 (TSTT=81404 with 0.7616% improvement and TSTT/TFFT=7.5339). RS surprisingly converges with high and sustained speed and achieves its best solution that is a little bit worse than GA1 at trial 227 (TSTT=81358.10 with 0.8175% improvement and TSTT/TFFT=7.5296). We suspect that the calibrated parameter set based on Network 1 may not be good enough for Network 4.1. Thus, we test another GA run (GA2) with the same parameter set and the addition of elitism. Interestingly, GA2 performs better than GA1. GA2 converges with high and sustained speed to find its best solution at trial 268 (TSTT=81328.76 with 0.8533% improvement

and $TSTT/TFFT=7.5269$). This implies metaheuristic parameters are network specific, and they should be calibrated for each network to obtain the best result.

It is observed that SA takes the longest CPU time for Networks 3.1 and 4.1 because the boundary constraint (Equation 4.12) and the budget constraint are not satisfied for many times and the regenerations (Steps 2-4) are required. It is noted that the results of SA on Networks 3.1 and 4.1 lead to the development of MSA discussed in Section 4.5, which is also calibrated and tested in Section 6.2 on Network 1. Based on the three criteria and the three test problems, GA outperforms SA and RS, and SA and RS appear non-dominated. In the next section, RS, MSA, GA, RTS-MT0, RTS-MT1 and RTS-MT2 are tested for the deterministic, stochastic and robust problems of NDP, SSD and NDP-SSD on Networks 3.2 and 4.2 in order to compare their performances and to identify a winner.

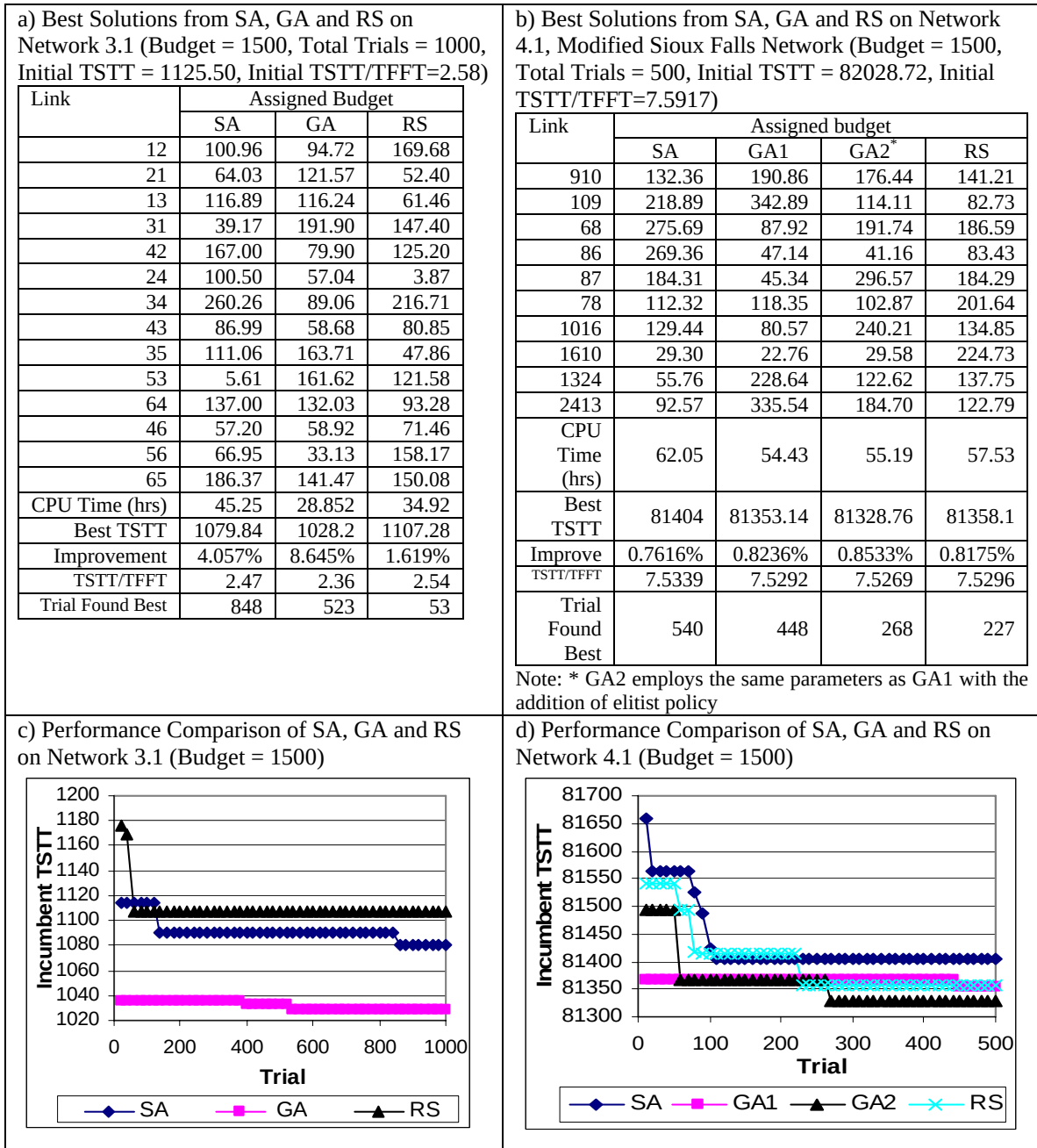


Figure 6.5. Results of Preliminary Performance Comparisons of SA, GA and RS for Deterministic NDP on Networks 3.1 and 4.1

6.4 Comprehensive Performance Comparisons of RS, MSA, GA, RTS-MT0, RTS-MT1 and RTS-MT2 on Networks 3.2 and 4.2

RS, MSA, GA, RTS-MT0, RTS-MT1 and RTS-MT2 are tested on Networks 3.2 and 4.2 for the nine problems:

- Deterministic SSD, NDP and NDP-SSD (DET-SSD, DET-NDP and DET-NDP-SSD)
- Stochastic SSD, NDP and NDP-SSD (SP-SSD, SP-NDP and SP-NDP-SSD)
- Robust SSD, NDP and NDP-SSD (RO-SSD, RO-NDP and RO-NDP-SSD)

MSA and GA employ the calibrated algorithm parameters in Figure 6.2. However, we also employ the elitist policy in GA as it makes GA perform better in the preliminary study in Section 6.3.2.

For the deterministic SSD, NDP and NDP-SSD, Network 3.2 with 72 O-D pairs is employed. Network 3.2 with 12 O-D pairs is used for the stochastic and robust problems with the O-D demand sample size of 5. Network 4.2 with 33 O-D pairs is employed for all nine problems, and for stochastic and robust problems, we use the O-D demand sample size of 3. The generated O-D demand samples are independent and identically distributed from the distribution of O-D demands. It is noted that the small sample size is assumed sufficient for comparing the performance of metaheuristics, and the best metaheuristic will be tested with a larger sample size (say 20) to obtain a solution in the next section.

To be comparable, for all iterations of all metaheuristics for stochastic and robust problems, the same set of O-D demand realizations is utilized. For the three SSD problems, all metaheuristics optimize the SSD variables with the original link capacity to obtain the best signal settings. Then, the three best signal settings obtained from deterministic, stochastic and robust SSD are fixed for running the respective deterministic, stochastic and robust NDP. This allows us to compare the sequential SSD and NDP against the combined NDP-SSD approach. For a fair comparison, the number of trials (i.e. the number of functional evaluations) for NDP-SSD is two times as many as that for SSD and NDP. That is, given the same total trials, the sequential approach is

compared with the combined approach. For the robust problems, we set $TSTT^{TARGET}$ equals to the minimal $E[TSTT]$ obtained from the sequential stochastic approach (SP-SSD and SP-NDP) and the combined approach (SP-NDP-SSD).

The performances of metaheuristics are compared according to solution quality and convergence speed. Unlike the preliminary study in the previous section, the CPU time cannot be compared because it highly depends on how busy the server was, and we had no control on other users of the server while conducting the experiment. Also, for the stochastic and robust problems, a trial involves performing multiple DTA runs corresponding to all O-D demand realizations. That is, the CPU time is dominated by the functional evaluation time, so the CPU times of all metaheuristics for the same total trials are approximately the same.

The common stopping criterion of the six metaheuristics is the total trials. For Networks 3.2, we employ 500 trials for DET-SSD and DET-NDP; 1000 trials for DET-NDP-SSD; 200 trials for SP-SSD, SP-NDP, RO-SSD and RO-NDP; and 400 trials for SP-NDP-SSD and RO-NDP-SSD. For Networks 4.2, we employ 250 trials for DET-SSD and DET-NDP; 500 trials for DET-NDP-SSD; 150 trials for SP-SSD, SP-NDP, RO-SSD and RO-NDP; and 300 trials for SP-NDP-SSD and RO-NDP-SSD. The summary results for the deterministic, stochastic and robust problems on Networks 3.2 and 4.2 are shown in Tables 6.4-6.6, and Figures 6.6-6.8 depicts the convergence characteristics in terms of incumbent objective value (i.e. best objective value found so far). Since the employed total trials are relatively small, especially for the stochastic and robust problems, the better solution quality also implies the better convergence speed. Thus, in this particular study, these two criteria may be considered the same, and we only discuss the comparison in terms of solution quality as this is of more interest.

We rank the six metaheuristics with respect to the solution quality for deterministic, stochastic and robust problems on Networks 3.2 and 4.2 in Tables 6.7-6.9. Apparently, there is not the single winning metaheuristic that outperforms the others across all nine problems on the two networks. For the deterministic problems, GA, RTS-MT2 and RS seem to be best, second best and third best, respectively. For the stochastic

problems, RTS-MT2, RS and GA appear best, second best and third best. For the robust problems, RTS-MT2, RS and GA are best, second best and third best. When considering all nine problems on the two networks, RTS-MT2 appears best as it is ranked in the top three except that it is ranked the fourth for RO-NDP on Network 4.2. GA is the second best as it is ranked in the top three except that it is the fourth for SP-NDP-SSD on Network 4.2, for RO-SSD on Network 4.2 and for RO-NDP-SSD on Network 3.2. RS is the third best as it is ranked in the top three except that it is the fourth for DET-SSD on Network 4.2, the fourth for SP-NDP-SSD on Network 3.2, and the fifth for DET-NDP-SSD on Network 3.2. RTS-MT0 performs the worst in all deterministic, stochastic and robust problems on both networks as it is always ranked in the bottom two. MSA performs the second worst as it is mostly ranked in the bottom three for all problems on both networks. RTS-MT1 can be considered the third worst as its performance is very random across all problems on both networks; for example, for RO-NDP-SSD, it is ranked best on Network 3.2 and the second worst on Network 4.2.

In addition, we compare the best objective values from the sequential approach and the integrated approach on Networks 3.2 and 4.2. We find that two out of six cases that the integrated approach yields better solutions than the sequential approach, namely, DET-NDP-SSD versus DET-NDP on Network 3.2 and RO-NDP-SSD versus RO-NDP on Network 4.2. However, this experiment may be considered biased toward the sequential approach because we use all six metaheuristics to find the best signal timing plans before running the six metaheuristics to find the best link improvement policy. A fairer comparison is conducted in the next section where only the identified best metaheuristics are employed in SSD and NDP.

Table 6.4. Summary Results of Six Metaheuristics for Deterministic SSD, NDP, NDP-SSD
a) Network 3.2 (72 O-D pairs)

MH	CPU (hours)	Best Objective Value (TSTT)	Trial Found
Deterministic SSD; 500 Trials			
RS	65.723	6431.82	94
MSA	63.4119	6647.80	463
GA	64.8560	6439.33	497
RTS-MT0	76.8773	7292.68	482
RTS-MT1	66.65	6846.04	488
RTS-MT2	65.2435	6475.59	289
Deterministic NDP; 500 Trials (with fixed best SSD solution from Deterministic SSD)			
RS	11.6296	6236.09	92
MSA	17.0460	6262.66	484
GA	15.24	6240.22	223
RTS-MT0	14.9979	6313.08	364
RTS-MT1	14.8133	6485.97	489
RTS-MT2	11.5578	6207.53	185
Deterministic NDP-SSD; 1000 Trials			
RS	141.8842	6385.04	411
MSA	161.6573	6335.81	619
GA	165.7906	6196.23	993
RTS-MT0	131.5626	6400.26	917
RTS-MT1	139.7382	6314.92	979
RTS-MT2	163.3615	6298.52	437

b) Network 4.2 (33 O-D pairs)

MH	CPU (hours)	Best Objective Value (TSTT)	Trial Found
Deterministic SSD; 250 Trials			
RS	16.8883	3615.26	103
MSA	17.0152	3707.16	47
GA	16.981	3555.26	214
RTS-MT0	16.5334	3747.80	130
RTS-MT1	16.2113	3574.78	234
RTS-MT2	16.8869	3540.97	139
Deterministic NDP; 250 Trials (with fixed best SSD solution from Deterministic SSD)			
RS	20.1622	3491.80	234
MSA	18.3969	3508.47	123
GA	20.4881	3456.23	136
RTS-MT0	21.9769	3512.28	250
RTS-MT1	21.9226	3554.85	167
RTS-MT2	20.4124	3488.69	214
Deterministic NDP-SSD; 500 Trials			
RS	38.719	3534.76	347
MSA	39.5309	3566.82	291
GA	38.6603	3483.68	89
RTS-MT0	37.6023	3779.38	497
RTS-MT1	39.0413	3923.07	464
RTS-MT2	39.2836	3511.30	274

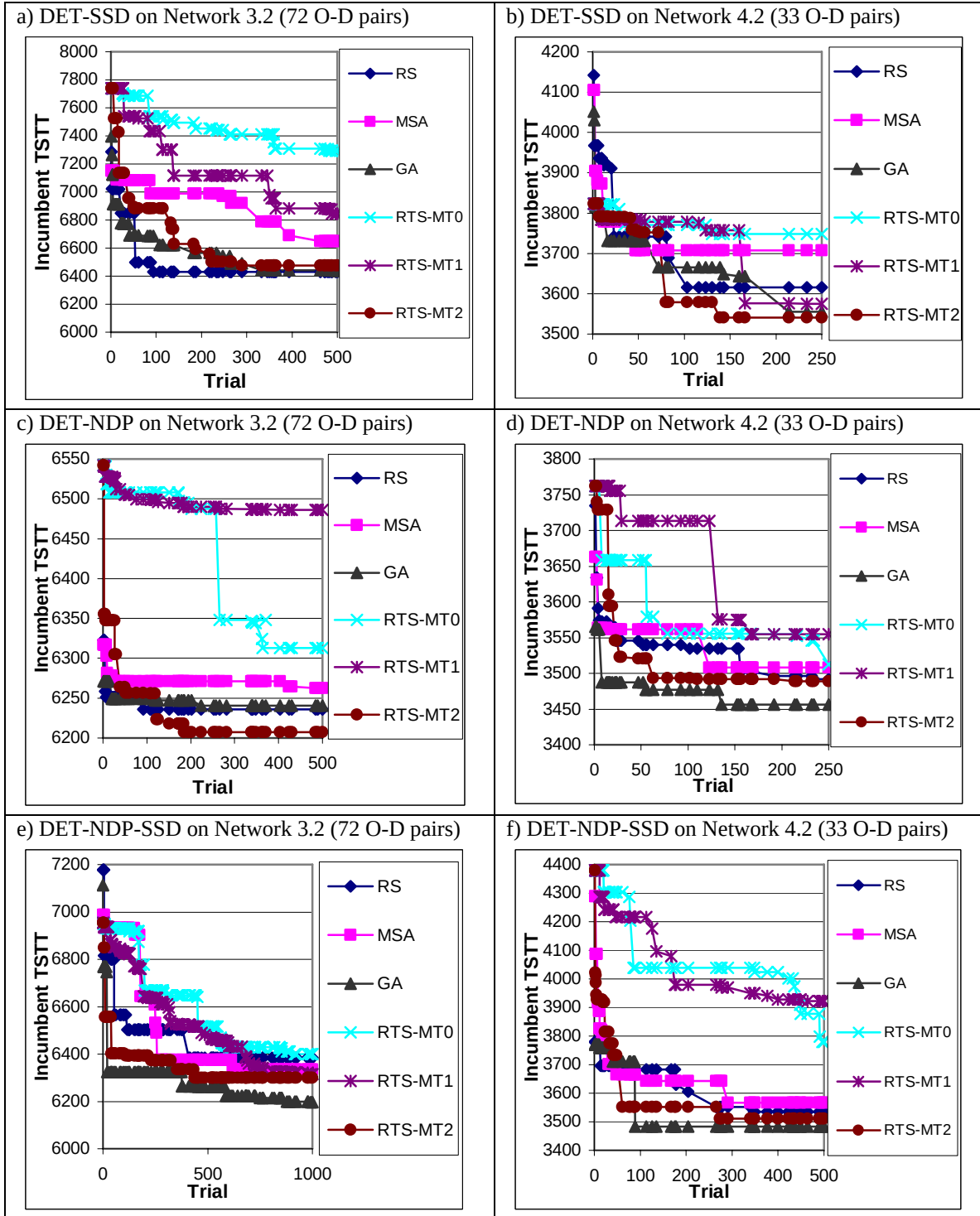


Figure 6.6. Convergence Characteristics of Metaheuristics for Deterministic Problems on Networks 3.2 and 4.2

Table 6.5. Results of Six Metaheuristics for Stochastic SSD, NDP, NDP-SSD

a) Network 3.2 (12 O-D pairs); Sample Size = 5

MH	CPU (hours)	Best Objective Value ($E[TSTT]$)	Trial Found
Stochastic SSD; 200 Trials			
RS	15.6304	583.69	110
MSA	14.68	623.09	172
GA	18.8539	584.09	173
RTS-MT0	15.0085	722.02	197
RTS-MT1	15.5805	596.81	184
RTS-MT2	18.5149	578.59	122
Stochastic NDP; 200 Trials (with fixed best SSD solution from Stochastic SSD)			
RS	20.4766	575.77	60
MSA	20.4578	575.85	155
GA	19.8309	575.77	77
RTS-MT0	20.5065	578.28	120
RTS-MT1	20.4952	581.03	199
RTS-MT2	20.5315	575.74	183
Stochastic NDP-SSD; 400 Trials			
RS	69.0568	582.70	158
MSA	69.1215	591.59	229
GA	68.6606	582.27	387
RTS-MT0	63.4991	585.07	288
RTS-MT1	63.9032	581.04	400
RTS-MT2	67.8907	578.18	381

b) Network 4.2 (33 O-D pairs); Sample Size = 3

MH	CPU (hours)	Best Objective Value ($E[TSTT]$)	Trial Found
Stochastic SSD; 150 Trials			
RS	35.4498	3563.63	83
MSA	34.7018	3711.89	6
GA	34.2636	3604.53	67
RTS-MT0	33.6107	3684.46	36
RTS-MT1	34.9729	3624.58	150
RTS-MT2	34.5239	3451.20	83
Stochastic NDP; 150 Trials (with fixed best SSD solution from Stochastic SSD)			
RS	43.2922	3420.46	77
MSA	43.2758	3430.94	48
GA	42.7902	3408.18	74
RTS-MT0	43.5742	3444.89	150
RTS-MT1	42.9431	3444.85	84
RTS-MT2	42.5331	3405.59	64
Stochastic NDP-SSD; 300 Trials			
RS	84.0314	3557.24	22
MSA	86.1259	3581.25	194
GA	86.5907	3600.27	197
RTS-MT0	76.7137	3828.70	85
RTS-MT1	86.0273	3610.30	298
RTS-MT2	84.999	3435.10	103

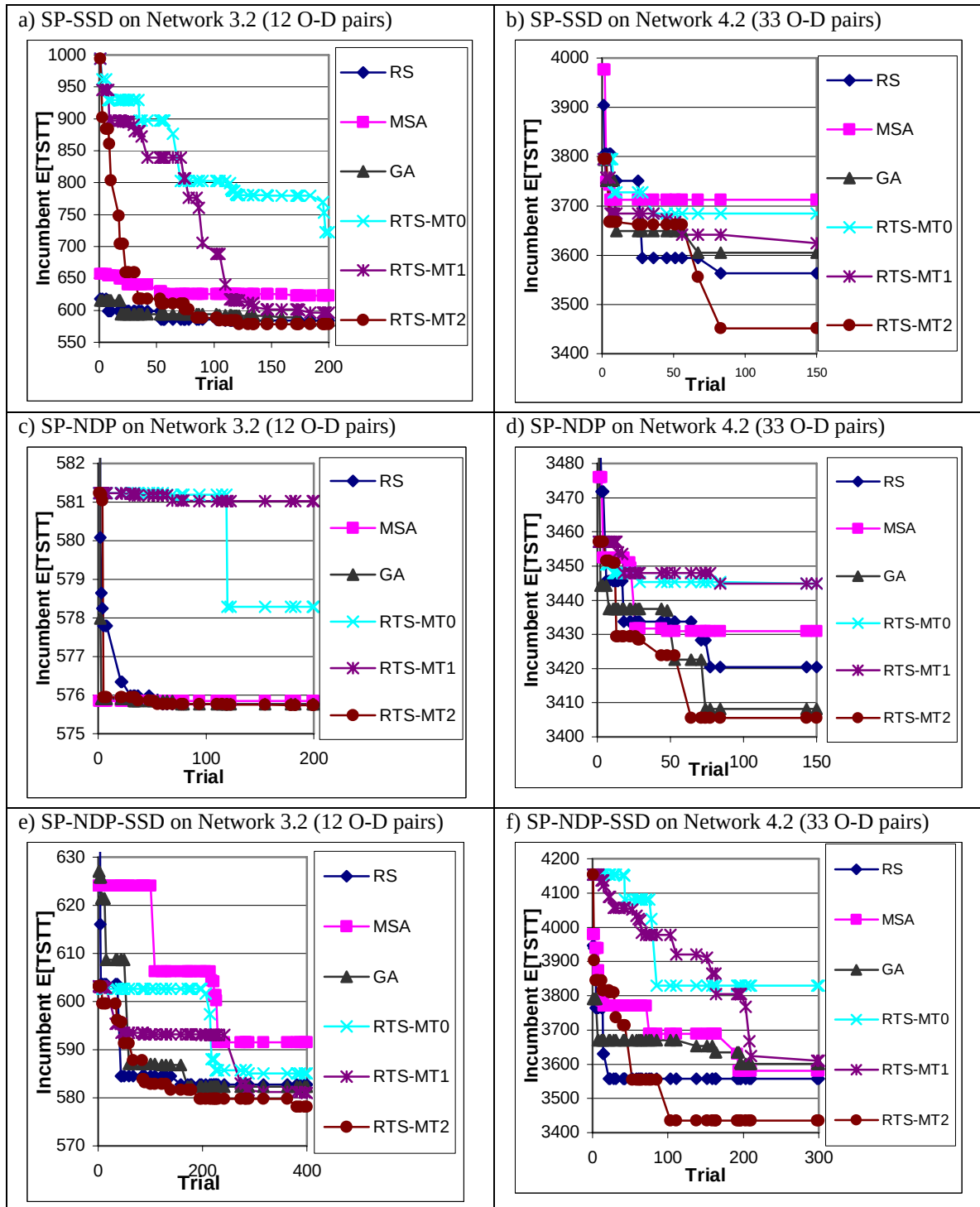


Figure 6.7. Convergence Characteristics of Metaheuristics for Stochastic Problems on Networks 3.2 and 4.2

Table 6.6. Summary Results of Six Metaheuristics for Robust SSD, NDP, NDP-SSDa) Network 3.2 (12 O-D pairs); Sample Size = 5; $\lambda = 1$; $TSTT^{TARGET} = 575.7428$

MH	CPU(hours)	Best Objective	$E[TSTT]$	$E[Risk]$	Trial Found
Robust SSD; 200 Trials					
RS	18.3583	921.85	585.394	336.456	54
MSA	18.5437	3385.76	620.102	2765.66	133
GA	18.2242	1054.01	586.544	467.469	109
RTS-MT0	18.3764	24550.03	722.022	23828	197
RTS-MT1	18.581	1686.81	603.011	1083.8	194
RTS-MT2	18.4818	928.96	585.407	343.554	83
Robust NDP; 200 Trials (with fixed best SSD solution from Robust SSD)					
RS	20.3013	801.69	580.916	220.777	171
MSA	20.3733	814.11	581.259	232.851	138
GA	20.302	801.25	580.463	220.782	182
RTS-MT0	20.3094	810.05	580.863	229.185	183
RTS-MT1	20.315	803.32	581.614	221.701	79
RTS-MT2	20.3521	800.34	580.403	219.933	119
Robust NDP-SSD; 400 Trials					
RS	51.2051	885.47	582.7	302.768	158
MSA	49.1172	1171.85	588.24	583.609	388
GA	48.1609	1108.81	595.139	513.675	210
RTS-MT0	44.2557	1112.47	588.004	524.471	217
RTS-MT1	43.3843	815.99	579.056	236.943	369
RTS-MT2	46.7279	889.96	583.19	306.769	286

b) Network 4.2 (33 O-D pairs); Sample Size = 3; $\lambda = 1$; $TSTT^{TARGET} = 3405.5906$

MH	CPU(hours)	Best Objective	$E[TSTT]$	$E[Risk]$	Trial Found
Robust SSD; 150 Trials					
RS	38.1939	31757.40	3563.63	28193.7	83
MSA	37.2715	102398.00	3711.89	98686.4	6
GA	37.5388	43284.10	3604.5261	39679.573	67
RTS-MT0	35.9166	82184.90	3684.4572	78500.403	36
RTS-MT1	38.0387	40990.47	3594.635	37395.834	136
RTS-MT2	37.6051	30464.85	3537.8427	26927.004	130
Robust NDP; 150 Trials (with fixed best SSD solution from Robust SSD)					
RS	35.7543	21288.70	3527.08	17761.6	80
MSA	36.0467	22085.30	3439.08	18646.2	40
GA	35.8158	21181.00	3524.9361	17656.039	114
RTS-MT0	35.9979	23873.00	3534.7317	20338.229	64
RTS-MT1	35.0966	25048.40	3536.6122	21511.789	145
RTS-MT2	36.0268	22685.32	3524.6494	19160.671	51
Robust NDP-SSD; 300 Trials					
RS	77.9798	26871.00	3557.24	23313.8	22
MSA	77.9003	31178.00	3559.66	27618.3	244
GA	78.2394	28763.21	3555.7917	25207.421	145
RTS-MT0	74.1122	190614.40	3828.6983	186785.71	85
RTS-MT1	77.8126	46394.12	3610.3011	42783.821	298
RTS-MT2	77.6504	9477.91	3435.0989	6042.8137	103

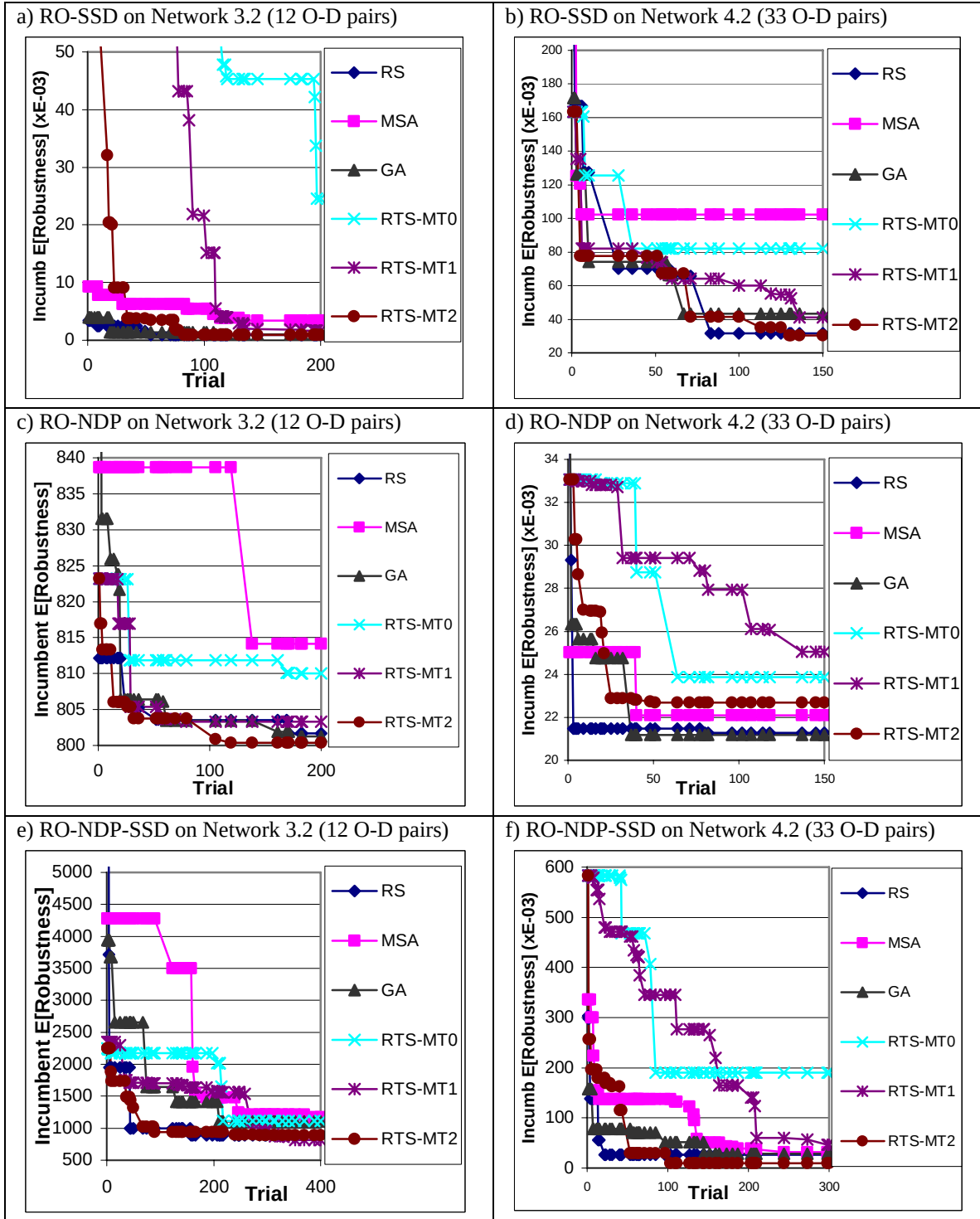


Figure 6.8. Convergence Characteristics of Metaheuristics for Robust Problems on Networks 3.2 and 4.2

Table 6.7. Rankings of Metaheuristics Based on Solution Quality for Deterministic Problems

a) Deterministic SSD

DET-SSD				
Network 3.2			Network 4.2	
Ranking	MH	Best Objective Value	MH	Best Objective Value
1	RS	6431.82	RTS-MT2	3540.97
2	GA	6439.33	GA	3555.26
3	RTS-MT2	6475.59	RTS-MT1	3574.78
4	MSA	6647.8	RS	3615.26
5	RTS-MT1	6846.04	MSA	3707.16
6	RTS-MT0	7292.68	RTS-MT0	3747.8

b) Deterministic NDP

DET-NDP				
Network 3.2			Network 4.2	
Ranking	MH	Best Objective Value	MH	Best Objective Value
1	RTS-MT2	6207.53	GA	3456.23
2	RS	6236.09	RTS-MT2	3488.69
3	GA	6240.22	RS	3491.8
4	MSA	6262.66	MSA	3508.47
5	RTS-MT0	6313.08	RTS-MT0	3512.28
6	RTS-MT1	6485.97	RTS-MT1	3554.85

c) Deterministic NDP-SSD

DET-NDP-SSD				
Network 3.2			Network 4.2	
Ranking	MH	Best Objective Value	MH	Best Objective Value
1	GA	6196.23	GA	3483.68
2	RTS-MT2	6298.52	RTS-MT2	3511.3
3	RTS-MT1	6314.92	RS	3534.76
4	MSA	6335.81	MSA	3566.82
5	RS	6385.04	RTS-MT0	3779.38
6	RTS-MT0	6400.26	RTS-MT1	3923.07

Table 6.8. Rankings of Metaheuristics Based on Solution Quality for Stochastic Problems

a) Stochastic SSD

	SP-SSD			
	Network 3.2		Network 4.2	
Ranking	MH	Best Objective Value	MH	Best Objective Value
1	RTS-MT2	578.59	RTS-MT2	3451.2
2	RS	583.69	RS	3563.63
3	GA	584.09	GA	3604.53
4	RTS-MT1	596.81	RTS-MT1	3624.58
5	MSA	623.09	RTS-MT0	3684.46
6	RTS-MT0	722.02	MSA	3711.89

b) Stochastic NDP

	SP-NDP			
	Network 3.2		Network 4.2	
Ranking	MH	Best Objective Value	MH	Best Objective Value
1	RTS-MT2	575.74	RTS-MT2	3405.59
2	RS	575.77	GA	3408.18
3	GA	575.77	RS	3420.46
4	MSA	575.85	MSA	3430.94
5	RTS-MT0	578.28	RTS-MT1	3444.85
6	RTS-MT1	581.03	RTS-MT0	3444.89

c) Stochastic NDP-SSD

	SP-NDP-SSD			
	Network 3.2		Network 4.2	
Ranking	MH	Best Objective Value	MH	Best Objective Value
1	RTS-MT2	578.18	RTS-MT2	3435.1
2	RTS-MT1	581.04	RS	3557.24
3	GA	582.27	MSA	3581.25
4	RS	582.7	GA	3600.27
5	RTS-MT0	585.07	RTS-MT1	3610.3
6	MSA	591.59	RTS-MT0	3828.7

Table 6.9. Rankings of Metaheuristics Based on Solution Quality for Robust Problems

a) Robust SSD

RO-SSD				
Network 3.2			Network 4.2	
Ranking	MH	Best Objective Value	MH	Best Objective Value
1	RS	921.85	RTS-MT2	30464.85
2	RTS-MT2	928.96	RS	31757.4
3	GA	1054.01	RTS-MT1	40990.47
4	RTS-MT1	1686.81	GA	43284.1
5	MSA	3385.76	RTS-MT0	82184.9
6	RTS-MT0	24550.03	MSA	102398

b) Robust NDP

RO-NDP				
Network 3.2			Network 4.2	
Ranking	MH	Best Objective Value	MH	Best Objective Value
1	RTS-MT2	800.34	GA	21181
2	GA	801.25	RS	21288.7
3	RS	801.69	MSA	22085.3
4	RTS-MT1	803.32	RTS-MT2	22685.32
5	RTS-MT0	810.05	RTS-MT0	23873
6	MSA	814.11	RTS-MT1	25048.4

c) Robust NDP-SSD

RO-NDP-SSD				
Network 3.2			Network 4.2	
Ranking	MH	Best Objective Value	MH	Best Objective Value
1	RTS-MT1	815.99	RTS-MT2	9477.91
2	RS	885.47	RS	26871
3	RTS-MT2	889.96	GA	28763.21
4	GA	1108.81	MSA	31178
5	RTS-MT0	1112.47	RTS-MT1	46394.12
6	MSA	1171.85	RTS-MT0	190614.4

6.5 Robustness Evaluation of Deterministic, Stochastic and Robust Solutions on Networks 3.2 and 4.2

Based on the results in the previous section, the identified best metaheuristic algorithm for each problem on each network is employed to find the near-global optimal solution. Since we would like to evaluate and compare the deterministic, stochastic and robust solutions, the same number of total trials is employed. Specifically, for Network 3.2 (12 O-D pairs), we employ 100 total trials for SSD and NDP problems; and 200 total trials for NDP-SSD problems. For Network 4.2 (33 O-D pairs), we employ 75 total trials for SSD and NDP problems and 150 total trials for NDP-SSD problems. For the

stochastic and robust problems on both networks, we generate 20 O-D demand realizations that are independent and identically distributed from the distribution of the O-D demand. Tables F.3 and F.5 (in Appendix F) show the 20 O-D demand realizations for Networks 3.2 and 4.2, respectively. For a fair comparison, this same set of 20 O-D demand realizations is used for all iterations of the metaheuristics for all stochastic and robust problems. For the deterministic problems, we employ the average demands.

For each network, we first run the metaheuristics for DET-NDP-SSD and SP-NDP-SSD to obtain the solutions of the integrated problems. Then, we run the metaheuristics for DET-SSD and SP-SSD to obtain the best signal settings. With the best signal settings, we solve DET-NDP and SP-NDP to obtain the solutions for the sequential SSD and NDP problems. The value of $TSTT^{TARGET}$ for the robust problems is set to the minimal value of $E[TSTT]$ from the sequential SP-SSD and SP-NDP approach and the integrated approach (SP-NDP-SSD). Subsequently, we run the metaheuristics for RO-SSD and RO-NDP-SSD. With the best signal settings from RO-SSD, we solve RO-NDP to obtain the solution of the sequential RO-SSD and RO-NDP approach. It is noted that the desirable λ value in robust problems depends on the preference of transportation planners, and the sensitivity analysis of λ value similar to Figures 5.8c-5.8d can be utilized to help elicit the decision makers' preference information. Since we use a larger sample size of O-D demands (i.e. 20 samples), a run of metaheuristic search is highly computationally expensive. Thus, we arbitrarily use $\lambda=1$. It is also noted that we tradeoff the O-D demand sample size (the larger sample size implies the better quality of solution) for the reasonable computational time. According to the rule of thumb in the Monte Carlo-based stochastic optimization (Morton, 2004), we use the sample size of 20. The larger sample size is indeed desirable; however, it costs too long computational time, given that we have to do multiple metaheuristic runs.

Since DET-NDP, SP-NDP and RO-NDP employ different traffic signal settings, these three may not be comparable. As such, we employ the original signal settings in Table 6.10 for NDP problems, and we denote the three NDP problems by NDP-D (where D stands for the default signal settings). Specifically, three additional NDP problems are

solved (DET-NDP-D, SP-NDP-D and RO-NDP-D). The detailed best solutions found by the metaheuristics for all twelve problems (i.e., DET/SP/RO SSD/NDP/NDP-D/NDP-SSD) are given in Appendix G. As can be seen from Appendix G, all deterministic, stochastic and robust SSD and NDP-SSD solutions employ different time offsets and different cycle lengths for intersections on both networks. This means we benefit from relaxing the common cycle length assumption and from the traffic signal coordination such that the great number of vehicles is progressed through the network.

To evaluate and compare all solutions from the twelve problems on Network 3.2 and 4.2, we generate a separate set of 30 O-D demand realizations that are independent and identically distributed from the distribution of O-D demands. Tables =F.4 and F.6 (in Appendix F) show the 30 O-D demand realizations for Networks 3.2 and 4.2, respectively. Evaluating a solution involves running a DTA corresponding to an O-D demand realization. After all 30 DTA runs, we compute the expected value, standard deviation, error and 95% confidence interval (CI) of $Robust_Obj(\lambda=1)$, $TSTT$ and $Risk$ from Equations (6.1)-(6.4). It is noted that the new set of 30 O-D demand realizations is different from the 20 O-D demand realizations used in the metaheuristic search in order to eliminate bias.

$$E[TSTT] = \frac{1}{N} \cdot \sum_{\omega=1}^N TSTT^{\omega} \quad (6.1.1)$$

$$E[Risk] = \frac{1}{N} \cdot \sum_{\omega=1}^N \left(\max(0, TSTT^{\omega} - TSTT^{TARGET}) \right)^2 \quad (6.1.2)$$

$$E[Robust_Obj(\lambda)] = E[TSTT] + \lambda \times E[Risk] \quad (6.1.3)$$

$$SD[TSTT] = \left(\frac{1}{N-1} \cdot \sum_{\omega=1}^N \left(TSTT^{\omega} - E[TSTT] \right)^2 \right)^{0.5} \quad (6.2.1)$$

$$SD[Risk] = \left(\frac{1}{N-1} \cdot \sum_{\omega=1}^N \left(Risk^{\omega} - E[Risk] \right)^2 \right)^{0.5} \quad (6.2.2)$$

$$SD[Robust_Obj(\lambda)] = \left(\frac{1}{N-1} \cdot \sum_{\omega=1}^N \left(Robust_Obj(\lambda)^{\omega} - E[Robust_Obj(\lambda)] \right)^2 \right)^{0.5} \quad (6.2.3)$$

$$Error[TSTT] = t_{\alpha/2, N-1} \cdot SD[TSTT] / \sqrt{N} \quad (6.3.1)$$

$$Error[Risk] = t_{\alpha/2, N-1} \cdot SD[Risk] / \sqrt{N} \quad (6.3.2)$$

$$Error[Robust_Obj(\lambda)] = t_{\alpha/2, N-1} \cdot SD[Robust_Obj(\lambda)] / \sqrt{N} \quad (6.3.3)$$

$$(1-\alpha)100\%CI \text{ of } E[TSTT] = (E[TSTT]-Error[TSTT], E[TSTT]+Error[TSTT]) \quad (6.4.1)$$

$$(1-\alpha)100\%CI \text{ of } E[Risk] = (E[Risk]-Error[Risk], E[Risk]+Error[Risk]) \quad (6.4.2)$$

$$(1-\alpha)100\%CI \text{ of } E[Robust_Obj(\lambda)] = (E[Robust_Obj(\lambda)]-Error[Robust_Obj(\lambda)], \\ E[Robust_Obj(\lambda)]+Error[Robust_Obj(\lambda)]) \quad (6.4.3)$$

We use $\alpha = 0.05$ and $N = 30$, so the corresponding student-t statistics are $t_{\alpha/2, N-1} = 2.045$. The detailed evaluation results of the original networks and the solutions for all twelve problems are given in Appendix H. Figures 6.9, 6.10 and 6.11 depict the 95% CIs of $E[Robust_Obj(\lambda=1)]$, $E[TSTT]$ and $E[Risk]$ for the original network and all twelve problems (DET/SSD/RO of SSD/NDP/NDP-D/NDP-SSD) on Networks 3.2 and 4.2. Figure 6.12 shows $V[TSTT]$ for the original network and the twelve problems on the two networks.

The comparisons of point estimates are conducted in three dimensions. First, we compare among the deterministic, stochastic and robust solutions on the two networks; that is, the solutions within the following three cases are compared against each other within the case: case 1 (DET-SSD, SP-SSD, RO-SSD), case 2 (DET-NDP-D, SP-NDP-D, RO-NDP-D) and case 3 (DET-NDP-SSD, SP-NDP-SSD, RO-NDP-SSD). We find that the RO solutions yield the least $E[Robust_Obj(\lambda=1)]$ and $E[Risk]$ as expected, whereas the DET solutions perform the worst with the highest $E[Robust_Obj(\lambda=1)]$, $E[TSTT]$ and $E[Risk]$. Since the metaheuristics for robust problems are not designed to minimize the $V[TSTT]$, the solutions of RO problems do not always yield the least $V[TSTT]$. For example, in case 1 (DET-SSD, SP-SSD, RO-SSD) on Network 4.2, the SP-SSD solution yields the least $V[TSTT]$, whereas the RO-SSD solution yields the least $E[Risk]$.

Second, we compare the sequential SSD and NDP approaches against the integrated NDP-SSD approaches; i.e., DET-NDP versus DET-NDP-SSD; SP-NDP versus SP-NDP-SSD; and RO-NDP versus RO-NDP-SSD. In terms of $E[Robust_Obj(\lambda=1)]$, the integrated approach performs better than the sequential approach except the case of

DET-NDP-SSD and DET-NDP on Network 4.2. In terms of $E[TSTT]$, the integrated approach performs better than the sequential approach in all cases. In terms of $E[Risk]$, the integrated approach outperforms the sequential approach in all cases except the case of DET-NDP-SSD versus DET-NDP on Network 4.2.

Third, we compare the original network and the twelve problems altogether. Apparently, the RO-NDP-SSD solution outperforms the others on both networks in terms of $E[Robust_Obj(\lambda=1)]$, $E[TSTT]$, $E[Risk]$ and $V[TSTT]$, whereas the sequential robust approach (i.e. RO-SSD & RO-NDP) performs the second best. The solutions from the twelve problems perform better than the original network in terms of $E[Robust_Obj(\lambda=1)]$, $E[TSTT]$ and $E[Risk]$ except DET-SSD on Network 3.2 and DET-NDP-D on Network 4.2. This shows that the deterministic approach can yield worse solutions than the do-nothing case when evaluated under stochastic condition. Thus, the integrated robust approach (RO-NDP-SSD) should be desirable whenever possible. Unlike the analytical result in Chapter 5, the sequential robust approach (RO-SSD & RO-NDP) performs the second best. This implies that for larger-size multi-destination networks where the analytical approach is not applicable, the integrated robust approach is still the most desirable; otherwise, the next best approach is the sequential robust approach.

Table 6.10. Original Traffic Signal Settings

a) Network 3.2

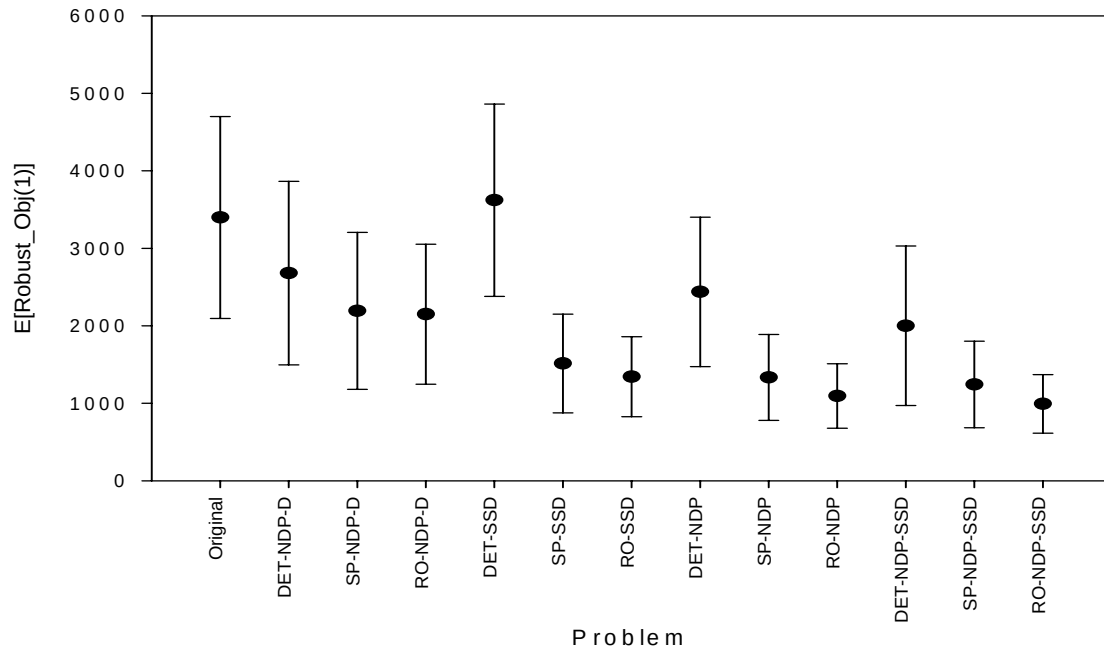
	Original Signal Setting
Cycle: Node 5	111
Node 4	94
Node 2	60
Offset: Node 5	11
Node 4	4
Node 2	26
Phase Seq: Node 5	6
Node 4	0
Node 2	1
Phase Green: Node 5	25, 17, 11, 34
Node 4	36, 15, 25
Node 2	17, 11, 14

b) Network 4.2

	Original Signal Setting
Cycle: Node 10	132
Node 8	84
Node 16	84
Node 6	63
Node 9	63
Node 17	63
Node 18	63
Offset: Node 10	11
Node 8	4
Node 16	26
Node 6	15
Node 9	9
Node 17	5
Node 18	20
Phase Seq: Node 10	0
Node 8	0
Node 16	0
Node 6	0
Node 9	0
Node 17	0
Node 18	0
Phase Green: Node 10	25, 17, 11, 34, 15
Node 8	15, 15, 15, 15
Node 16	15, 15, 15, 15
Node 6	15, 15, 15
Node 9	15, 15, 15
Node 17	15, 15, 15
Node 18	15, 15, 15

Note: All red time is 2 seconds per movement phase, and yellow time is 4 seconds per movement phase.

a) Network 3.2



b) Network 4.2

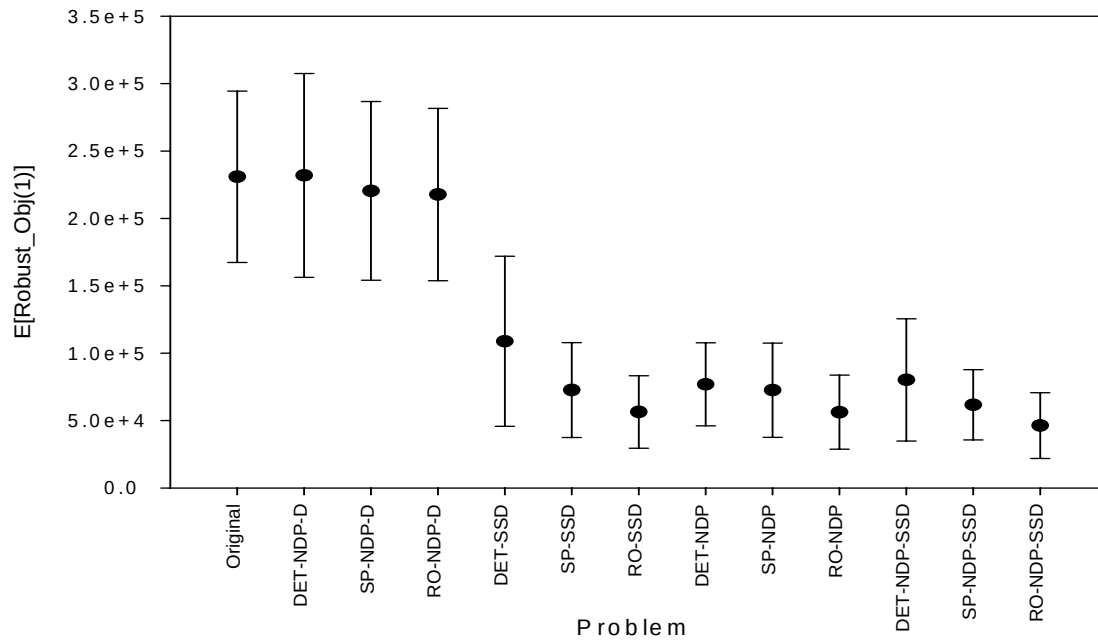
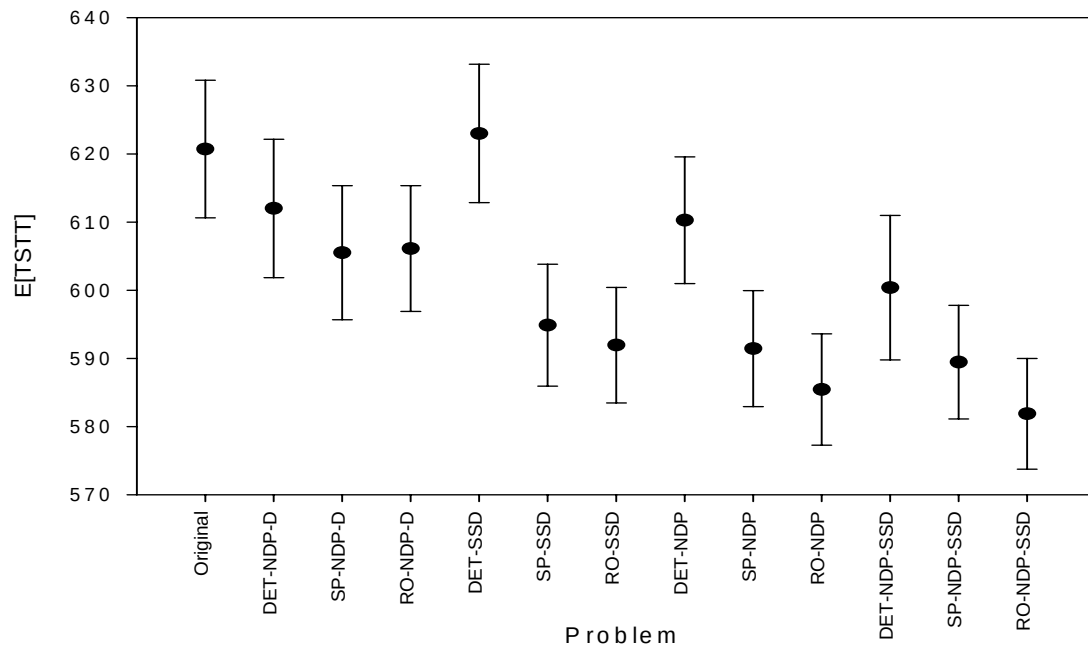


Figure 6.9. 95% Confidence Intervals of $E[\text{Robust_Obj}(\lambda=1)]$

a) Network 3.2



b) Network 4.2

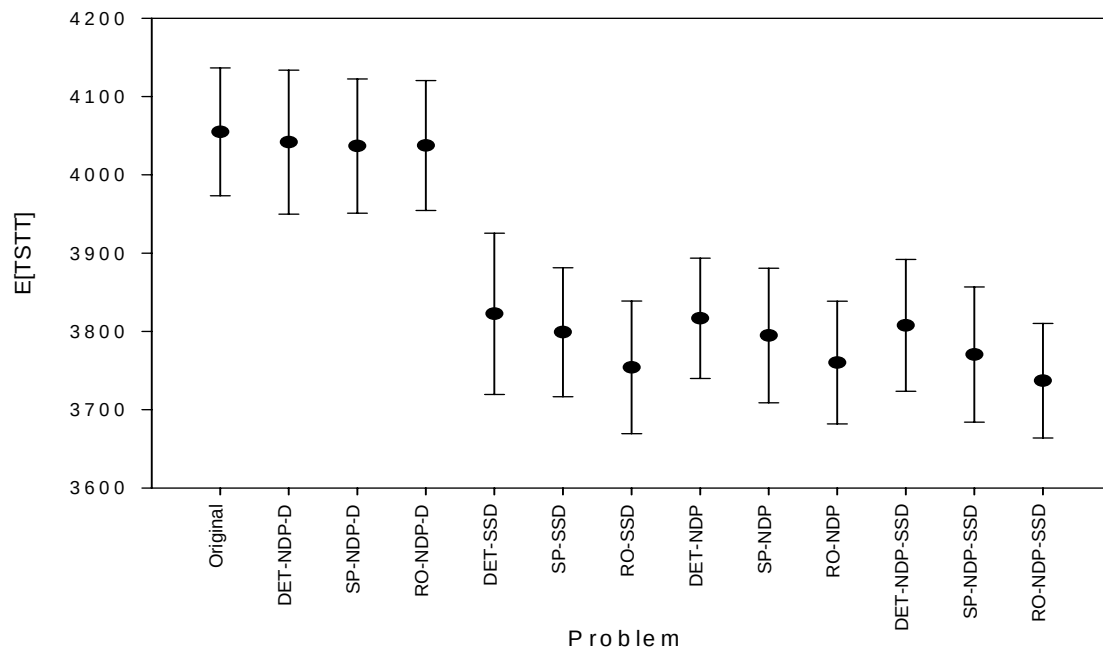
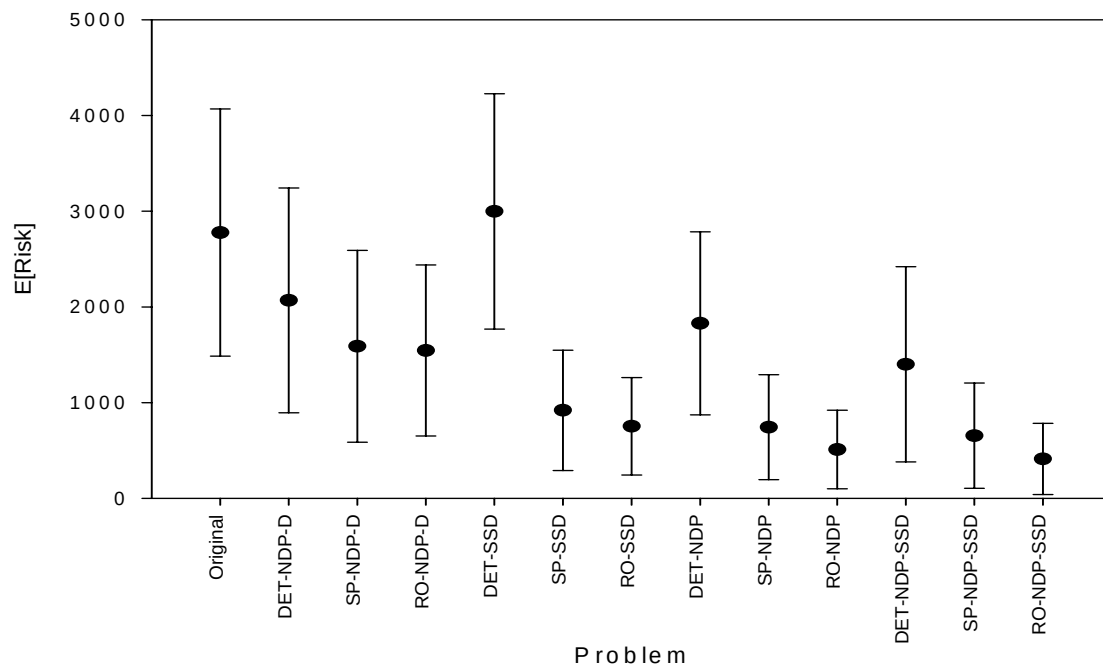


Figure 6.10. 95% Confidence Intervals of $E[TSTT]$

a) Network 3.2



b) Network 4.2

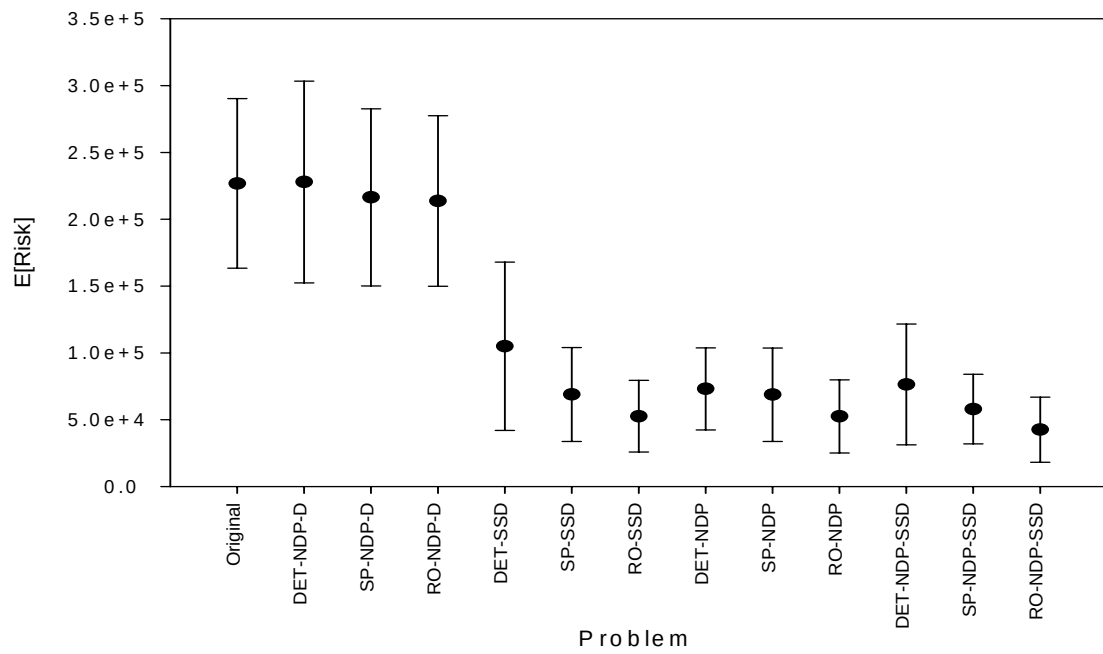
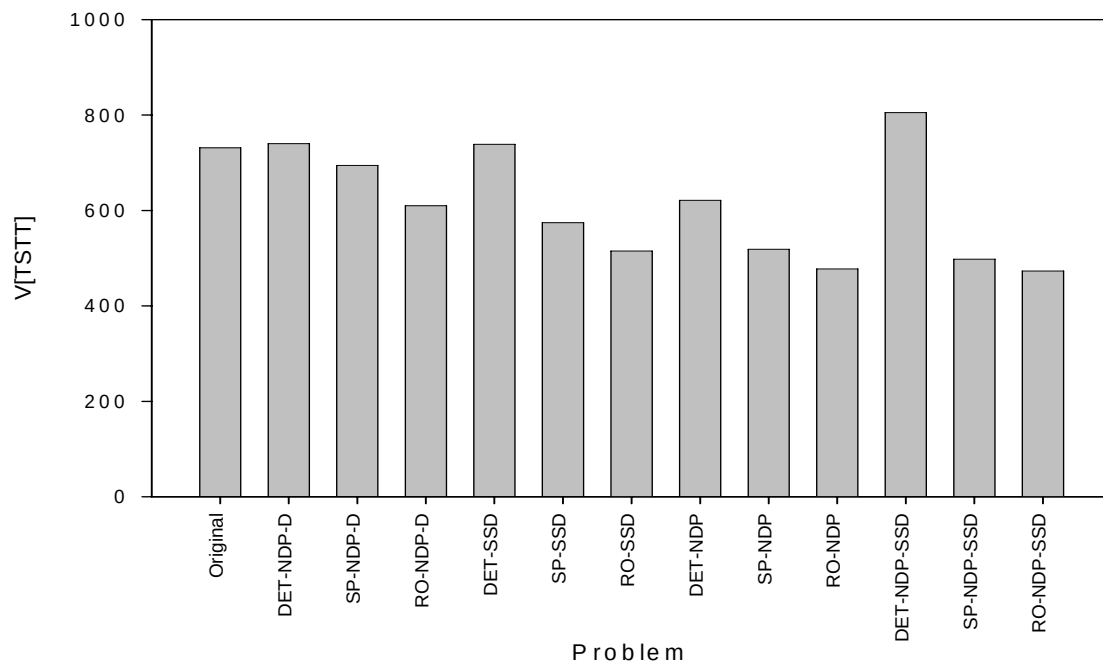


Figure 6.11. 95% Confidence Intervals of $E[\text{Risk}]$

a) Network 3.2



b) Network 4.2

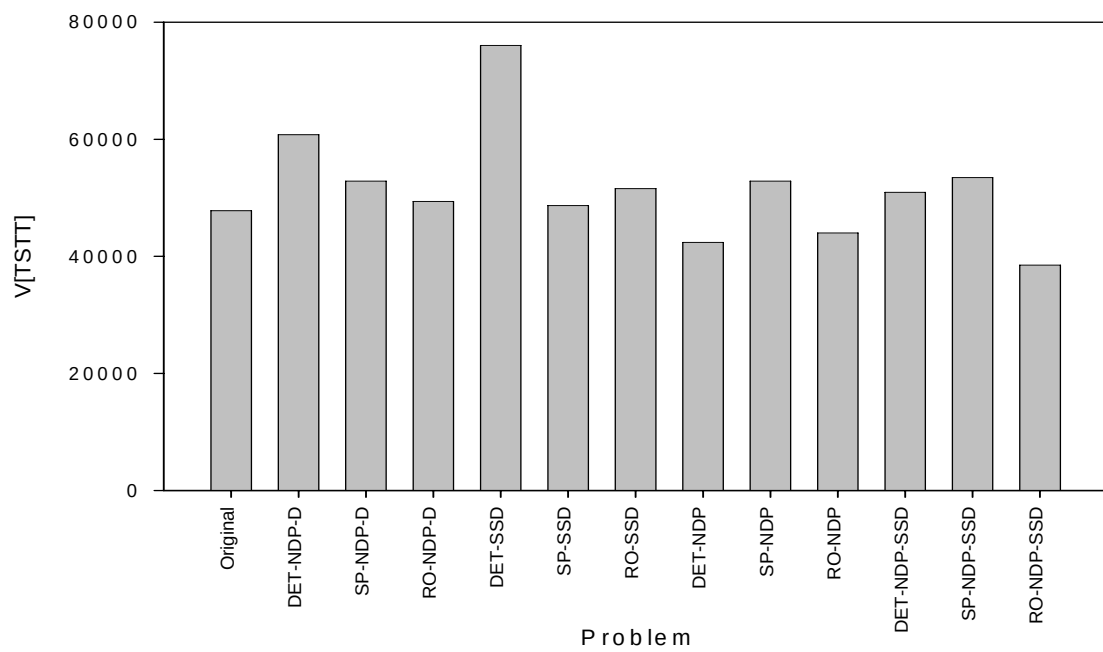


Figure 6.12. Variance of TSTT

In the previous paragraphs, we made rather weak comparisons that one solution is better than another in expected values. Apparently, one can observe the overlapping 95% confidence intervals of $E[TSTT]$, $E[Risk]$ and $E[Robust_Obj(1)]$ of many solutions. One can make a stronger comparison via paired t tests. We perform 36 paired t tests to test the null hypotheses that the RO-NDP-SSD solution outperforms the other 12 solutions in terms of the three measures ($E[TSTT]$, $E[Risk]$ and $E[Robust_Obj(1)]$).

For illustration, let us consider the first paired t test, which tests the null hypothesis that $E[TSTT]$ of the RO-NDP-SSD solution is better than $E[TSTT]$ of the original network. Let $D_{TSTT} = TSTT_{RO-NDP-SSD} - TSTT_{original}$ be the difference of $TSTT$ corresponding to an arbitrary O-D demand realization. Then,

$$\begin{aligned}\mu_{D_{TSTT}} &= E[TSTT_{RO-NDP-SSD} - TSTT_{original}] = E[TSTT_{RO-NDP-SSD}] - E[TSTT_{original}] \\ &= \mu_{TSTT_{RO-NDP-SSD}} - \mu_{TSTT_{original}}.\end{aligned}$$

Since $D_{i,TSTT} = TSTT_{i,RO-NDP-SSD} - TSTT_{i,original}$ for $i=1,2,...,n$ constitute a normal random sample (of differences) with mean $\mu_{D_{TSTT}}$, the hypothesis about $\mu_{D_{TSTT}}$ can be tested using a one-sample t test based on $n-1$ degree of freedom (Devore, 2000). Specifically, the null hypothesis and the alternative hypothesis are:

$$H_0 : \mu_{D_{TSTT}} = E[TSTT_{RO-NDP-SSD}] - E[TSTT_{original}] \leq 0$$

$$H_a : \mu_{D_{TSTT}} = E[TSTT_{RO-NDP-SSD}] - E[TSTT_{original}] > 0$$

The test statistic value is:

$$t = \frac{\bar{d}_{TSTT} - 0}{s_{D_{TSTT}} / \sqrt{n}}.$$

where $\bar{d}_{TSTT}$ and $s_{D_{TSTT}}$ are the sample mean and standard deviation, respectively, of the $D_{i,TSTT}$'s. Also, the corresponding p -value can be determined from the t test statistic. The rejection region for level α test is:

$$t \geq t_{\alpha, n-1} \text{ (i.e. } p\text{-value} \leq \alpha \text{)}.$$

In addition, a confidence interval for $\mu_{D_{TSTT}}$ can be constructed. The $100(1-\alpha)\%$ confidence interval for $\mu_{D_{TSTT}}$ is:

$$\bar{d}_{TSTT} \pm t_{\alpha/2, n-1} \cdot s_{D_{TSTT}} / \sqrt{n}$$

It is noted that we use $n=30$, so the central limit theorem guarantees that the resulting interval is valid without any restrictions on the distribution of the differences (Devore, 2000).

TSTT, *Risk* and *Robust_Obj(1)* of 13 solutions corresponding to the 30 O-D demand realizations on Network 3.2 are shown in Tables I.1-I.3 in Appendix I, and those on Network 4.2 are shown in Tables I.4-I.6 in Appendix I. The *TSTT* differences, *Risk* differences and *Robust_Obj(1)* differences between the RO-NDP-SSD solution and the other 12 solutions for the 30 O-D demand realizations are shown in Tables I.7-I.9 and I.10-I.12 for Networks 3.2 and 4.2, respectively. Also, the corresponding 95% confidence intervals of $E[TSTT]$, $E[Risk]$ and $E[Robust_Obj(1)]$ are shown in Tables I.1-I.6.

Table 6.11 summarizes the paired t tests on the null hypothesis that the $E[TSTT]$ of the RO-NDP-SSD solution is better than that of the other 12 solutions on Networks 3.2 and 4.2. Similarly, Tables 6.12 and 6.13 summarize the paired t tests on the null hypotheses that the $E[Risk]$ and $E[Robust_Obj(1)]$ of the RO-NDP-SSD solution are better than those of the other 12 solutions on the two networks. Apparently, the minimal p -value from the 72 paired t tests (36 paired t tests for each network) is 0.794997. Thus, the results of all 72 paired t tests imply that on the two networks we fail to reject the null hypothesis when $\alpha < 0.794997$.

Table 6.11. Paired t Tests of $TSTT$ Differences

a) Network 3.2

Test	$\bar{d}_{TSTT}$	$s_{D_{TSTT}}$	t	p -value
D(13-1)	-38.84	16.68	-12.755	1.000000
D(13-2)	-30.13	13.76	-11.993	1.000000
D(13-3)	-23.64	12.69	-10.206	1.000000
D(13-4)	-24.25	11.79	-11.261	1.000000
D(13-5)	-41.13	11.75	-19.169	1.000000
D(13-6)	-13.00	6.82	-10.436	1.000000
D(13-7)	-10.07	6.94	-7.949	1.000000
D(13-8)	-28.40	11.41	-13.632	1.000000
D(13-9)	-9.56	8.91	-5.878	0.999999
D(13-10)	-3.55	6.11	-3.184	0.998274
D(13-11)	-18.50	10.78	-9.395	1.000000
D(13-12)	-7.57	7.74	-5.358	0.999995

b) Network 4.2

Test	$\bar{d}_{TSTT}$	$s_{D_{TSTT}}$	t	p -value
D(13-1)	-317.91	111.62	-15.600	1.000000
D(13-2)	-304.73	151.20	-11.039	1.000000
D(13-3)	-299.76	129.45	-12.683	1.000000
D(13-4)	-300.45	125.80	-13.082	1.000000
D(13-5)	-85.51	202.58	-2.312	0.985963
D(13-6)	-62.02	103.64	-3.278	0.998640
D(13-7)	-17.03	111.56	-0.836	0.794997
D(13-8)	-79.72	166.03	-2.630	0.993237
D(13-9)	-57.80	116.96	-2.707	0.994365
D(13-10)	-23.13	100.58	-1.260	0.891100
D(13-11)	-70.69	132.44	-2.924	0.996677
D(13-12)	-33.43	155.09	-1.181	0.876342

Note: D(13-x) is the $TSTT$ difference of RO-NDP-SSD and x, where the values of x are: 1=do-nothing; 2=DET-NDP-D; 3=SP-NDP-D; 4=RO-NDP-D; 5=DET-SSD; 6=SP-SSD; 7=RO-SSD; 8=DET-NDP; 9=SP-NDP; 10=RO-NDP; 11=DET-NDP-SSD; 12=SP-NDP-SSD; 13=RO-NDP-SSD

Table 6.12. Paired t Tests of *Risk Differences*

a) Network 3.2

Test	$\bar{d}_{Risk}$	$s_{D_{Risk}}$	t	p -value
D(13-1)	-2366.03	2551.44	-5.079	0.999990
D(13-2)	-1657.21	2264.11	-4.009	0.999805
D(13-3)	-1176.76	1784.52	-3.612	0.999433
D(13-4)	-1133.07	1489.43	-4.167	0.999873
D(13-5)	-2586.86	2410.94	-5.877	0.999999
D(13-6)	-508.17	750.45	-3.709	0.999562
D(13-7)	-340.76	458.71	-4.069	0.999834
D(13-8)	-1416.80	1668.41	-4.651	0.999967
D(13-9)	-331.73	557.79	-3.257	0.998568
D(13-10)	-97.82	264.71	-2.024	0.973863
D(13-11)	-988.77	1783.14	-3.037	0.997495
D(13-12)	-243.05	568.21	-2.343	0.986895

b) Network 4.2

Test	$\bar{d}_{Risk}$	$s_{D_{Risk}}$	t	p -value
D(13-1)	-184256.09	141052.07	-7.155	1.000000
D(13-2)	-185319.78	174747.96	-5.809	0.999999
D(13-3)	-173793.55	147489.98	-6.454	1.000000
D(13-4)	-171110.97	141950.29	-6.602	1.000000
D(13-5)	-62433.17	160396.51	-2.132	0.979197
D(13-6)	-26285.35	77290.87	-1.863	0.963669
D(13-7)	-10059.04	61911.53	-0.890	0.809581
D(13-8)	-30523.41	69095.19	-2.420	0.988982
D(13-9)	-26181.00	79294.36	-1.808	0.959541
D(13-10)	-9905.62	62146.22	-0.873	0.805090
D(13-11)	-33837.26	111492.45	-1.662	0.946387
D(13-12)	-15393.83	57789.66	-1.459	0.922345

Note: D(13-x) is the *Risk difference* of RO-NDP-SSD and x, where the values of x are: 1=do-nothing; 2=DET-NDP-D; 3=SP-NDP-D; 4=RO-NDP-D; 5=DET-SSD; 6=SP-SSD; 7=RO-SSD; 8=DET-NDP; 9=SP-NDP; 10=RO-NDP; 11=DET-NDP-SSD; 12=SP-NDP-SSD; 13=RO-NDP-SSD

Table 6.13. Paired t Tests of $Robust_Obj(1)$ Differences

a) Network 3.2

Test	$\bar{d}_{Robust_Obj(1)}$	$s_{D_{Robust_Obj(1)}}$	t	p -value
D(13-1)	-2404.87	2560.92	-5.143	0.999991
D(13-2)	-1687.34	2273.39	-4.065	0.999833
D(13-3)	-1200.40	1792.31	-3.668	0.999512
D(13-4)	-1157.32	1495.76	-4.238	0.999896
D(13-5)	-2627.99	2418.41	-5.952	0.999999
D(13-6)	-521.17	754.07	-3.786	0.999643
D(13-7)	-350.83	462.14	-4.158	0.999870
D(13-8)	-1445.20	1674.88	-4.726	0.999973
D(13-9)	-341.29	562.15	-3.325	0.998798
D(13-10)	-101.37	269.23	-2.062	0.975876
D(13-11)	-1007.26	1791.62	-3.079	0.997746
D(13-12)	-250.62	571.87	-2.400	0.988490

b) Network 4.2

Test	$\bar{d}_{Robust_Obj(1)}$	$s_{D_{Robust_Obj(1)}}$	t	p -value
D(13-1)	-184574.04	141123.18	-7.164	1.000000
D(13-2)	-185624.51	174858.58	-5.814	0.999999
D(13-3)	-174093.34	147577.89	-6.461	1.000000
D(13-4)	-171411.65	142033.25	-6.610	1.000000
D(13-5)	-62518.67	160563.00	-2.133	0.979228
D(13-6)	-26347.41	77376.72	-1.865	0.963838
D(13-7)	-10076.19	61994.88	-0.890	0.809665
D(13-8)	-30603.20	69214.85	-2.422	0.989036
D(13-9)	-26238.85	79386.92	-1.810	0.959690
D(13-10)	-9928.82	62227.21	-0.874	0.805332
D(13-11)	-33908.04	111608.89	-1.664	0.946563
D(13-12)	-15427.37	57899.28	-1.459	0.922401

Note: D(13-x) is the $Robust_Obj(1)$ difference of RO-NDP-SSD and x, where the values of x are: 1=do-nothing; 2=DET-NDP-D; 3=SP-NDP-D; 4=RO-NDP-D; 5=DET-SSD; 6=SP-SSD; 7=RO-SSD; 8=DET-NDP; 9=SP-NDP; 10=RO-NDP; 11=DET-NDP-SSD; 12=SP-NDP-SSD; 13=RO-NDP-SSD

6.6. Summary

This chapter presents the computational experiences of the seven metaheuristics: SA, MSA, GA, RTS-MT0, RTS-MT1, RTS-MT2 and RS. First, it describes four multi-origin multi-destination test networks. Next, the implementation of all metaheuristics and the calibration of SA, MSA and GA parameters for deterministic NDP are described. Then, with the calibrated SA, MSA and GA algorithm parameters, the performance comparison of the seven metaheuristics on Network 1 shows that RTS-MT1 outperformed the other algorithms based on the three criteria: solution quality, convergence speed and CPU time. The second best are RTS-MT2 and GA. RTS-MT0 appears moderate, and the bottom three algorithms performing very poorly are RS, SA and MSA. Next, only SA, GA and RS are tested on Networks 3.1 and 4.1, and the results show that GA appears the best whereas RS and SA are non-dominated.

Subsequently, RS, MSA, GA, RTS-MT0, RTS-MT1 and RTS-MT2 are comprehensively compared for the deterministic, stochastic and robust problems of SSD, NDP and NDP-SSD on Networks 3.2 and 4.2. For the stochastic and robust problems, we used the O-D demand sample sizes of 5 and 3 for Networks 3.2 and 4.2. These O-D demand realizations are independent and identically distributed from the distribution of O-D demands. Since we cannot capture the true CPU time of each metaheuristic run and the functional evaluation time dominates the total CPU time, the CPU times of different metaheuristic algorithms are approximately the same. Also, since the used total trials are relatively small, especially for the stochastic and robust problems, the better solution quality also implies the better convergence speed. As such, we only discuss the comparison in terms of solution quality. The best metaheuristic algorithm is identified for each problem on each network. We found that there is not the single winner across the problems on both networks. For all nine problems, the metaheuristics ranked from best to worst are RTS-MT2, GA, RS, RTS-MT1, MSA and RTS-MT0.

Then, the identified best metaheuristic algorithm for each problem on each network is used to find the near-global optimal solution. For all stochastic and robust problems, we use the same set of 20 O-D demand realizations, which are independent and

identically distributed from the distribution of O-D demands. For the robust problems, we set the target TSTT equal to the best expected TSTT from the stochastic problems, and use $\lambda = 1$. All deterministic, stochastic and robust SSD and NDP-SSD solutions employ different time offsets and different cycle lengths for intersections on Networks 3.2 and 4.2. This implies that we benefit from the relaxation of the common cycle length assumption typically used in the literature and from the traffic signal coordination.

All solutions are evaluated under stochastic condition by using the new 30 O-D demand realizations in order to eliminate a possible bias. We evaluate the solutions in terms of $E[Robust_Obj(\lambda=1)]$, $E[TSTT]$, $E[Risk]$ and $V[TSTT]$ and also construct the 95% confidence intervals of $E[Robust_Obj(\lambda=1)]$, $E[TSTT]$ and $E[Risk]$. The comparisons of the robustness evaluation are conducted in three ways. First, when comparing between deterministic, stochastic and robust problems, the robust solutions yield the least $E[Robust_Obj(\lambda=1)]$ and $E[Risk]$, and the deterministic solutions perform the worst with the greatest $E[Robust_Obj(\lambda=1)]$, $E[Risk]$ and $E[TSTT]$. Since the metaheuristics for the robust problems are not designed to minimize the $V[TSTT]$, the solutions of RO problems do not always yield the least $V[TSTT]$. Second, when comparing the sequential SSD and NDP approach against the integrated NDP-SSD approach, in terms of $E[Robust_Obj(\lambda=1)]$, the integrated approach performs better than the sequential approach except the deterministic case on Network 4.2. In terms of $E[TSTT]$, the integrated approach performs better than the sequential approach in all cases. In terms of $E[Risk]$, the integrated approach outperforms the sequential approach in all cases except the deterministic case on Network 4.2. Third, when comparing all problems altogether, the integrated robust (RO-NDP-SSD) solution outperforms the others on both networks in terms of $E[Robust_Obj(\lambda=1)]$, $E[TSTT]$, $E[Risk]$ and $V[TSTT]$, whereas the sequential robust approach (i.e. RO-SSD & RO-NDP) performs the second best. The solutions from all problems perform better than the original network in terms of $E[Robust_Obj(\lambda=1)]$, $E[TSTT]$ and $E[Risk]$ except certain deterministic cases. This shows that the deterministic approach could yield worse solutions than the do-nothing scenario when evaluating under stochastic condition. Thus,

the integrated robust approach (RO-NDP-SSD) should be sought-after whenever possible. Unlike the analytical result in Chapter 5, the sequential robust approach (RO-SSD & RO-NDP) performs the second best. This implies that for larger-size multi-destination networks where the analytical approach is not applicable, the integrated robust approach is the most desirable; otherwise, the next best approach is the sequential robust approach.

Furthermore, we illustrate that a stronger comparison can be made via the paired t test. The 72 paired t tests are conducted to test the null hypotheses that the RO-NDP-SSD solution outperforms the other 12 solutions in $E[TSTT]$, $E[Risk]$ and $E[Robust_Obj(1)]$ on the two networks. The minimal p -value from the 72 paired t tests is 0.794997; thus, we fail to reject the null hypotheses when $\alpha < 0.794997$.

Chapter 7. Extensions, Summary and Conclusions

7.1 Extensions

Three possible extensions of this dissertation are discussed: 1) a mathematical programming-based heuristic for BLPNDP; 2) NDP, SSD and NDP-SSD formulations that incorporate O-D demand elasticity; and 3) other metaheuristic schemes.

7.1.1 Dantzig-Wolfe Decomposition-Based Heuristic Scheme for BLPNDP

The mathematical programming-based heuristic for BLPNDP is inspired from the decomposition techniques in large-scale linear programming and the work by Li et al. (2003) that decomposed the SO DTA LP model (the model 2.11) using the Dantzig-Wolfe decomposition approach (Dantzig, 1963). BLPNDP may not be decomposed due to the bi-level structure. However, it embeds the UODTA LP model as the lower-level program, so we may heuristically apply the decomposition scheme to BLPNDP to obtain a local optimal solution. Let us consider the SONDP (7.1) and its dual program (7.2):

Primal SONDP

$$\min_{x,y} \sum_{\forall t \in T} \sum_{\forall (i,j) \in E_s} t \cdot y_{ij}^t \quad (7.1.1a)$$

$$\sum_{i \in C \setminus C_S} b_i \leq TAB \quad : \rho \quad (7.1.2)$$

$$b_i \geq 0 \quad \forall i \in C \setminus C_S \quad (7.1.3)$$

$$x_i^t - x_i^{t-1} + \sum_{(i,j) \in FS(i)} y_{ij}^{t-1} - \sum_{(j,i) \in RS(i)} y_{ji}^{t-1} = d_i^t \quad \forall i \in C \setminus C_S, t \in T \quad : \pi_i^{0,t} \quad (7.1.4)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^t - x_i^t \leq 0 \quad \forall i \in C \setminus C_S, t \in T \quad : \pi_i^{1,t} \quad (7.1.5)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^t \leq \delta_i^t (N_i^t + \chi_i \cdot b_i - x_i^t) \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad : \pi_i^{2,t} \quad (7.1.5)$$

$$\sum_{(j,i) \in RS(i)} y_{ji}^t \leq Q_i^t + \phi_i \cdot b_i \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad : \pi_i^{3,t} \quad (7.1.6)$$

$$\sum_{(i,j) \in FS(i)} y_{ij}^t \leq Q_i^t + \phi_i \cdot b_i \quad \forall i \in C \setminus C_S, t \in T \quad : \pi_i^{4,t} \quad (7.1.7)$$

$$x_i^0 = \zeta_i \quad \forall i \in C \setminus C_S \quad (7.1.8)$$

$$y_{ij}^0 = 0 \quad \forall (i, j) \in E \quad (7.1.9)$$

$$x_i^{|T|} = 0 \quad \forall i \in C \setminus C_S \quad (7.1.10)$$

$$x_i^t \geq 0 \quad \forall i \in C \setminus C_S, t \in T \quad (7.1.11)$$

$$y_{ij}^t \geq 0 \quad \forall (i, j) \in E, t \in T \quad (7.1.12)$$

Dual SONDP

$$\max_{\pi} \sum_{\forall t \in T} \left(\sum_{\forall i \in C_R} d_i^t \pi_i^{0,t} - \sum_{\forall i \in C \setminus (C_R \cup C_S)} \delta_i^t N_i^t \pi_i^{2,t} - \sum_{\forall i \in C \setminus (C_R \cup C_S)} Q_i^t \pi_i^{3,t} - \sum_{\forall i \in C \setminus C_S} Q_i^t \pi_i^{4,t} \right) - \rho \cdot TAB \quad (7.2.1)$$

subject to

$$\pi_i^{0,t} - \pi_i^{0,t+1} + \pi_i^{1,t} \leq 0 \quad \forall i \in C_R, t \in T \setminus \{|T|\} \quad : x_i^t \quad (7.2.2)$$

$$\pi_i^{0,t} - \pi_i^{0,t+1} + \pi_i^{1,t} - \delta_i^t \pi_i^{2,t} \leq 0 \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \setminus \{|T|\} \quad : x_i^t \quad (7.2.3)$$

$$\pi_i^{0,t+1} - \pi_j^{0,t+1} - \pi_i^{1,t} - \pi_j^{2,t} - \pi_j^{3,t} - \pi_i^{4,t} \leq 0 \quad \forall (i, j) \in E \setminus E_S, t \in T \setminus \{|T|\} \quad : y_{ij}^t \quad (7.2.4)$$

$$\pi_i^{0,t+1} - \pi_i^{1,t} - \pi_i^{4,t} \leq t \quad \forall (i, j) \in E_S, t \in T \setminus \{|T|\} \quad : y_{ij}^t \quad (7.2.5a)$$

$$-\rho + \sum_{i \in C \setminus (C_R \cup C_S)} \sum_{t \in T} \delta_i^t \chi_i \pi_i^{2,t} + \sum_{i \in C \setminus (C_R \cup C_S)} \sum_{t \in T} \phi_i \pi_i^{3,t} + \sum_{i \in C \setminus C_S} \sum_{t \in T} \phi_i \pi_i^{4,t} \leq 0 \quad \forall i \in C \setminus C_S \quad : b_i \quad (7.2.6)$$

$$\pi_i^{0,t} \text{ u.r.s. } \forall i \in C \setminus C_S, t \in T \quad (7.2.7)$$

$$\pi_i^{1,t}, \pi_i^{4,t} \geq 0 \quad \forall i \in C \setminus C_S, t \in T \quad (7.2.8)$$

$$\pi_i^{2,t}, \pi_i^{3,t} \geq 0 \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \quad (7.2.9)$$

$$\rho \geq 0 \quad (7.2.10)$$

In the dual program of SONDP, there is a complicating constraint (Constraint 7.2.6), and if this is removed, the resulting program is simply the dual program of SODTA, which is a known problem. Thus, the Dantzig-Wolfe decomposition technique can be applied directly to the dual program of SONDP resulting in the full master program (7.3) and the pricing subproblem (7.4) (i.e. the dual program of SODTA).

Dantzig-Wolfe Full Master Program

$$\begin{aligned} & \max_{\pi} \\ & \sum_{v=1}^V \left(w_v \cdot \sum_{\forall t \in T} \left(\sum_{\forall i \in C_R} d_i^t \pi_i^{0,t,v} - \sum_{\forall i \in C \setminus (C_R \cup C_S)} \delta_i^t N_i^t \pi_i^{2,t,v} - \sum_{\forall i \in C \setminus (C_R \cup C_S)} Q_i^t \pi_i^{3,t,v} - \sum_{\forall i \in C \setminus C_S} Q_i^t \pi_i^{4,t,v} \right) \right) - \rho \cdot TAB \end{aligned} \quad (7.3.1)$$

subject to

$$\begin{aligned} & -\rho + \sum_{v=1}^V \left(w_v \cdot \left(\sum_{i \in C \setminus (C_R \cup C_S)} \sum_{t \in T} \delta_i^t \chi_i \pi_i^{2,t,v} + \sum_{i \in C \setminus (C_R \cup C_S)} \sum_{t \in T} \phi_i \pi_i^{3,t,v} + \sum_{i \in C \setminus C_S} \sum_{t \in T} \phi_i \pi_i^{4,t,v} \right) \right) \leq 0 \\ & \forall i \in C \setminus C_S \quad : b_i \end{aligned} \quad (7.3.2)$$

$$\sum_{v=1}^V w_v = 1 \quad : g \quad (7.3.3)$$

$$w_v \geq 0 \quad \forall v = 1, 2, \dots, V \quad (7.3.4)$$

$$\rho \geq 0 \quad (7.3.5)$$

Dantzig-Wolfe Subproblem (Pricing Problem)

$$\begin{aligned} & \max_{\pi_i^{0,t}, \pi_i^{1,t}, \pi_i^{2,t}, \pi_i^{3,t}, \pi_i^{4,t}} \sum_{\forall t \in T} \left(\sum_{\forall i \in C_R} d_i^t \pi_i^{0,t} - \sum_{\forall i \in C \setminus (C_R \cup C_S)} \delta_i^t N_i^t \pi_i^{2,t} - \sum_{\forall i \in C \setminus (C_R \cup C_S)} Q_i^t \pi_i^{3,t} - \sum_{\forall i \in C \setminus C_S} Q_i^t \pi_i^{4,t} \right) \\ & - \sum_{i \in C \setminus C_S} \left(b_i \cdot \left(\sum_{i \in C \setminus (C_R \cup C_S)} \sum_{t \in T} \delta_i^t \chi_i \pi_i^{2,t} + \sum_{i \in C \setminus (C_R \cup C_S)} \sum_{t \in T} \phi_i \pi_i^{3,t} + \sum_{i \in C \setminus C_S} \sum_{t \in T} \phi_i \pi_i^{4,t} \right) \right) - g \end{aligned} \quad (7.4.1)$$

$$\pi_i^{0,t} - \pi_i^{0,t+1} + \pi_i^{1,t} \leq 0 \quad \forall i \in C_R, t \in T \setminus \{|T|\} \quad : x_i^t \quad (7.4.2)$$

$$\pi_i^{0,t} - \pi_i^{0,t+1} + \pi_i^{1,t} - \delta_i^t \pi_i^{2,t} \leq 0 \quad \forall i \in C \setminus (C_R \cup C_S), t \in T \setminus \{|T|\} \quad : x_i^t \quad (7.4.3)$$

$$\begin{aligned} & \pi_i^{0,t+1} - \pi_j^{0,t+1} - \pi_i^{1,t} - \pi_j^{2,t} - \pi_i^{3,t} - \pi_i^{4,t} \leq 0 \quad \forall (i, j) \in E \setminus E_S, t \in T \setminus \{|T|\} \\ & \quad : y_{ij}^t \end{aligned} \quad (7.4.4)$$

$$\pi_i^{0,t+1} - \pi_i^{1,t} - \pi_i^{4,t} \leq t \quad \forall (i, j) \in E_S, t \in T \setminus \{|T|\} \quad : y_{ij}^t \quad (7.4.5)$$

$$\pi_i^{0,t} \text{ u.r.s. } \forall i \in C \setminus C_s, t \in T \quad (7.4.6)$$

$$\pi_i^{1,t}, \pi_i^{4,t} \geq 0 \quad \forall i \in C \setminus C_s, t \in T \quad (7.4.7)$$

$$\pi_i^{2,t}, \pi_i^{3,t} \geq 0 \quad \forall i \in C \setminus (C_R \cup C_s), t \in T \quad (7.4.8)$$

It is noted that the pricing problem is a modified UODTA by replacing the original Q_i^t and N_i^t with $Q_i^t + \phi_i b_i$ and $N_i^t + \chi_i b_i$ where b_i are given by the master program. We heuristically modify the Dantzig-Wolfe algorithm by changing the pricing problem from SODTA to UODTA. A possible heuristic algorithm for the deterministic NDP is the following.

Dantzig-Wolfe-Decomposition Based Algorithm for NDP

Data: $N_i^{t,original}, Q_i^{t,original}, d_i^t, \delta_i^t \quad \forall i \in C, t \in T; \chi_i, \phi_i \quad \forall i \in C \setminus C_s; TAB$

Step 0: Set $v = 1$;

Set $N_i^t = N_i^{t,original} \quad \forall i \in C, t \in T$;

Set $Q_i^t = Q_i^{t,original} \quad \forall i \in C, t \in T$;

Set $b_i = 0 \quad \forall i \in C \setminus C_s$

Set $g = 0$.

Step 1: Solve the combinatorial UODTA with N_i^t and $Q_i^t \quad \forall i \in C, t \in T$ to obtain $x_i^t \quad \forall i \in C \setminus C_s, t \in T$, $y_{ij}^t \quad \forall (i, j) \in E, t \in T$.

Step 2: Solve for the approximate dual variables based on the solution in Step 1 to obtain $\pi_i^{0,t,v}, \pi_i^{1,t,v}, \pi_i^{2,t,v}, \pi_i^{3,t,v}, \pi_i^{4,t,v}$.

Step 3: If the objective function value of the subproblem is less than or equal to zero,

stop. That is, if $\sum_{t \in T} \left(\sum_{i \in C_R} d_i^t \pi_i^{0,t,v} - \sum_{i \in C \setminus (C_R \cup C_s)} \delta_i^t N_i^t \pi_i^{2,t,v} - \sum_{i \in C \setminus (C_R \cup C_s)} Q_i^t \pi_i^{3,t,v} - \sum_{i \in C \setminus C_s} Q_i^t \pi_i^{4,t,v} \right)$

$$- \sum_{i \in C \setminus C_S} \left(b_i \cdot \left(\sum_{i \in C \setminus (C_R \cup C_S)} \sum_{t \in T} \delta_i^t \chi_i \pi_i^{2,t,v} + \sum_{i \in C \setminus (C_R \cup C_S)} \sum_{t \in T} \phi_i \pi_i^{3,t,v} + \sum_{i \in C \setminus C_S} \sum_{t \in T} \phi_i \pi_i^{4,t,v} \right) \right) - g \leq 0, \text{ stop}$$

and return the solution $b_i \ \forall i \in C \setminus C_S$, $x_i^t \ \forall i \in C \setminus C_S, t \in T$, $y_{ij}^t \ \forall (i, j) \in E, t \in T$.

Otherwise, go to Step 4.

Step 4: Solve the restricted master program by CPLEX to obtain the dual variables of the master program: $b_i \ \forall i \in C \setminus C_S$ and g .

Step 5: Set $v = v + 1$;

$$\text{Set } N_i^t = N_i^{t, \text{original}} + \chi_i \cdot b_i ;$$

$$\text{Set } Q_i^t = Q_i^{t, \text{original}} + \phi_i \cdot b_i .$$

Go to Step 1

In the proposed algorithm, the pricing problem is first solved by either mathematical programming (UO DTA LP for single-destination network), the combinatorial UO DTA algorithm (Waller, 2000), or the simulation-based approach (VISTA software by Ziliaskopoulos and Waller, 2000); then, the dual variables ($\pi_i^{0,t}$, $\pi_i^{1,t}$, $\pi_i^{2,t}$, $\pi_i^{3,t}$, $\pi_i^{4,t}$) have to be calculated from the primal solution as discussed below.

Dual Variables Approximation (Step 2)

In Step 2 of the heuristic algorithm, the variables in the pricing problem are calculated based on the known primal variables of UO DTA, and this calculation involves certain approximation. The approximation can be done in many ways, and the best approximation can be determined from the computational experiment. However, in this dissertation, we only propose the scheme without any experiment and leave this for the future research.

The dual variables ($\pi_i^{0,t}$, $\pi_i^{1,t}$, $\pi_i^{2,t}$, $\pi_i^{3,t}$, $\pi_i^{4,t}$) can approximately be determined from the primal variables (x_i^t , y_{ij}^t) by using the complementary slackness conditions and the problem-specific interpretation of the dual variables. Let us consider the time-

expanded network of the original network. When taking into account the flows (x_i^t, y_{ij}^t) , the time-expanded network with Q_i^t and N_i^t can be viewed as an acyclic graph that may or may not be connected. For example, a particular cell-time node (i, t) in the time-expanded network is blocked, if $N_i^t - x_i^t = 0$ or $Q_i^t - \sum_{(j,i) \in RS(i)} y_{ji}^t = 0$.

Let $TSTT_{original}$ denote the total system travel time of the original network. The dual variable $\pi_i^{0,t}$ can be interpreted as the change of the objective function (TSTT) when the demand of cell i at time t is increased by one unit. This is equivalent to the change of TSTT caused by adding a vehicle at cell i and time t . This additional vehicle follows the shortest path in the time-expanded network to the destination. This path can be efficiently determined from the time-expanded network by the search algorithm. Then, we run the simulation to calculate $TSTT_{new}$, and calculate $\pi_i^{0,t} = TSTT_{new} - TSTT_{original}$. By repeating this process for all cells and time intervals, we obtain $\pi_i^{0,t} \forall i \in C \setminus C_s, t \in T$.

Next, let's consider the complementary slackness conditions associated with (2.11.3)-(2.11.6). The dual variables are set to zero when the associated constraints are non-binding; otherwise, they are only known to be non-negative:

- If $\sum_{(i,j) \in FS(i)} y_{ij}^t - x_i^t < 0$, then set $\pi_i^{1,t} = 0 \forall i \in C \setminus C_s, t \in T$.
- If $\sum_{(j,i) \in RS(i)} y_{ji}^t < \delta_i^t (N_i^t - x_i^t)$, set $\pi_i^{2,t} = 0 \forall i \in C \setminus (C_R \cup C_s), t \in T$.
- If $\sum_{(j,i) \in RS(i)} y_{ji}^t < Q_i^t$, set $\pi_i^{3,t} = 0 \forall i \in C \setminus (C_R \cup C_s), t \in T$.
- If $\sum_{(i,j) \in FS(i)} y_{ij}^t < Q_i^t$, set $\pi_i^{4,t} = 0 \forall i \in C \setminus C_s, t \in T$.
- If $\sum_{(i,j) \in FS(i)} y_{ij}^t - x_i^t = 0$, then $\pi_i^{1,t} \geq 0 \forall i \in C \setminus C_s, t \in T$.
- If $\sum_{(j,i) \in RS(i)} y_{ji}^t = \delta_i^t (N_i^t - x_i^t)$, then $\pi_i^{2,t} \geq 0 \forall i \in C \setminus (C_R \cup C_s), t \in T$.
- If $\sum_{(j,i) \in RS(i)} y_{ji}^t = Q_i^t$, then $\pi_i^{3,t} \geq 0 \forall i \in C \setminus (C_R \cup C_s), t \in T$.

- If $\sum_{(i,j) \in FS(i)} y_{ij}^t = Q_i^t$, then $\pi_i^{4,t} \geq 0 \quad \forall i \in C \setminus C_s, t \in T$.

Then, the complementary slackness conditions for (2.12.2) are employed to determine $\pi_i^{1,t} \quad \forall i \in C_R, t \in T \setminus \{T\}$:

- If $x_i^t > 0$, then set $\pi_i^{1,t} = \pi_i^{0,t+1} - \pi_i^{0,t} \quad \forall i \in C_R, t \in T \setminus \{T\}$.
- Else if $x_i^t = 0$ and $\pi_i^{1,t}$ is undetermined, we approximate the inequality

$$\pi_i^{1,t} \leq \pi_i^{0,t+1} - \pi_i^{0,t} \quad \forall i \in C_R, t \in T \setminus \{T\} \text{ by equality.}$$

Subsequently, the complementary slackness conditions for (2.12.3) are utilized to determine $\pi_i^{1,t}$ and $\pi_i^{2,t} \quad \forall i \in C \setminus (C_R \cup C_s), t \in T \setminus \{T\}$:

- If $x_i^t > 0$, $\pi_i^{1,t} - \delta_i^t \pi_i^{2,t} = \pi_i^{0,t+1} - \pi_i^{0,t} \quad \forall i \in C \setminus (C_R \cup C_s), t \in T \setminus \{T\}$.
- Else if $x_i^t = 0$, $\pi_i^{1,t} - \delta_i^t \pi_i^{2,t} \leq \pi_i^{0,t+1} - \pi_i^{0,t} \quad \forall i \in C \setminus (C_R \cup C_s), t \in T \setminus \{T\}$.

We approximate the dual variables by treating the inequality as equality. If $\pi_i^{1,t}$ and $\pi_i^{2,t}$ are both undetermined, we assume $\pi_i^{1,t} = \pi_i^{2,t}$.

Next, the complementary slackness conditions for (2.12.4) are employed to determine $\pi_j^{3,t}$ and $\pi_i^{4,t} \quad \forall (i, j) \in E \setminus E_s, t \in T \setminus \{T\}$:

- If $y_{ij}^t > 0$, $\pi_j^{3,t} + \pi_i^{4,t} = \pi_i^{0,t+1} - \pi_j^{0,t+1} - \pi_i^{1,t} - \pi_j^{2,t} \quad \forall (i, j) \in E \setminus E_s, t \in T \setminus \{T\}$.
- Else if $y_{ij}^t = 0$, $\pi_j^{3,t} + \pi_i^{4,t} \geq \pi_i^{0,t+1} - \pi_j^{0,t+1} - \pi_i^{1,t} - \pi_j^{2,t}$

$$\forall (i, j) \in E \setminus E_s, t \in T \setminus \{T\}.$$

We approximate the dual variables by treating the inequality as equality. If $\pi_j^{3,t}$ and $\pi_i^{4,t}$ are both undetermined, we assume $\pi_j^{3,t} = \pi_i^{4,t}$.

Lastly, the complementary slackness conditions for (2.12.5a) are employed to determine $\pi_i^{4,t} \quad \forall (i, j) \in E_s, t \in T \setminus \{T\}$:

- If $y_{ij}^t > 0$, $\pi_i^{4,t} = \pi_i^{0,t+1} - \pi_i^{1,t} - t \quad \forall (i, j) \in E_s, t \in T \setminus \{T\}$.
- Else if $y_{ij}^t = 0$, $\pi_i^{4,t} \geq \pi_i^{0,t+1} - \pi_i^{1,t} - t \quad \forall (i, j) \in E_s, t \in T \setminus \{T\}$.

We approximate the dual variables by treating the inequality as equality.

7.1.2 Models Accounting for O-D Demand Elasticity

A common limitation of the NDP, SSD and NDP-SSD formulations in this study is the assumption of O-D demand inelasticity; that is, the demands are unchanged regardless how the system is improved. In fact, the O-D demands are elastic, and should be modified according to the changed trip costs, which are resulted from the network improvement decisions. A possible extension is the approximation of $d_i^t \forall i \in C_R, t \in T$ as the function of O-D trip costs (i.e. the function of $x_i^t \forall i \in C \setminus C_S, t \in T$ and $y_{ij}^t \forall (i, j) \in E, t \in T$). For the stochastic problems, the long-term demand uncertainty would involve the coefficients of the function of O-D trip costs being random variables. If the O-D demand functions are linear, the exact solution methods discussed in this dissertation are still applicable to the models that account for O-D demand elasticity.

7.1.3 Other Metaheuristics

In this dissertation, we explore a collection of metaheuristic algorithms; however, there are other promising metaheuristic algorithms in the literature that should be further investigated such as the ant colony algorithm (Dorigo and Stützle, 2004) and the continuous reactive tabu search algorithm (Battiti and Tecchiolli, 1996). Also, the hybrid metaheuristic algorithm may be developed. For example, from our findings, we know that RTS-MT1 performs best when the feasible region is relatively small (from Section 6.3.1), and RTS-MT2 and GA perform very well in the larger feasible region (from Section 6.4). A possible hybrid algorithm could be composed of two modes: diversification and intensification. The algorithm starts with the diversification mode that can be either GA or RTS-MT2 to quickly reach a good area within the feasible region; then, the algorithm switches to the intensification mode that employs RTS-MT0 to thoroughly search this area of the feasible region.

7.2 Summary and Conclusions

This dissertation proposes bi-level mathematical models under deterministic, two-stage stochastic, and robust optimization schemes for the three transportation network improvement optimization problems: continuous network design problem (NDP), DTA-based signal setting design problem (SSD) and the combined NDP, SSD and DTA problem (NDP-SSD). The exact solution methods are also proposed. Further, the four more practical metaheuristic methods, namely, simulated annealing (SA), genetic algorithm (GA), reactive tabu search (RTS) and random search (RS), are developed for solving the NDP, SSD and NDP-SSD problems under the deterministic, stochastic and robust optimization schemes. Based on the initial results on the deterministic NDP, SA is modified to improve efficiency, and the resulting algorithm is the modified simulated annealing (MSA). Further, three variations of RTS based on three move type definitions (or neighborhood definitions) are developed, namely, RTS-MT0, RTS-MT1 and RTS-MT2.

Literature Review

The review of the literature to provide a basis for understanding the models and algorithms proposed in this dissertation is described. It starts off by reviewing the literature on NDP, SSSD, and NDP-SSD. Then, cell transmission-based dynamic traffic assignment (DTA) procedures are discussed, including linear programming models and simulation-based approach. Next, five relevant optimization schemes are described, namely, stochastic optimization, robust optimization, linear bi-level optimization, zero-one continuous linear bi-level programming and metaheuristics.

Analytical Approach

Subsequently, a core set of models that can allow new perspectives on traditionally difficult and intractable NDP, SSD and NDP-SSD problems is developed. All three transportation network improvement optimization problems are bi-level by nature; i.e. the transportation planner makes different design decisions to optimize the system, whereas the transportation users react to the design decisions such that the dynamic user equilibrium is realized. Further, the problem complexity is increased by

considering the uncertainty of the time-dependent origin-destination (O-D) demands due to the inherent stochastic nature of forecasted traveler demand. As such, the problems are naturally modeled as two-stage stochastic programs. That is, the planners make their design decisions, considered as the first-stage decision variables, before the O-D demands are realized; subsequently, the transportation users make their route choice decisions, considered as the second-stage decision variables. Furthermore, to better deal with the uncertainty, not only the expected cost is included in the objective function but also the expected risk. Thus, the problems are modeled as the robust optimization programs. With such complexity these models are not developed for final applications and are limited to solving a single-destination, small network; however, they enable us to gain insights into the problems and facilitate the progressive development towards the practical metaheuristic methods.

Metaheuristic Approach

The more practical metaheuristic methods are developed in this dissertation, namely, SA, MSA, GA, RTS-MT0, RTS-MT1, RTS-MT2 and RS. For a clear presentation, the decision variables of NDP, SSD and NDP-SSD are explicitly specified. Then, the associated solution representations and decoding procedures with respect to different metaheuristics are described. Next, the common objective functional evaluation procedures for the deterministic, stochastic and robust optimization schemes are discussed. Subsequently, the formal metaheuristic algorithms are provided.

Experimental Results of Analytical Approach

Next, the preliminary results of the deterministic NDP models are presented using a cell transmission test network, Network 1. For the deterministic bi-level NDP model, the modified K^{th} best algorithm is employed without the guarantee of exact solutions. However, it intuitively provides a possible steepest-descent-type iterative heuristic for multi-destination networks. Since this involves the cumbersome pre-processing task on a simplex tableau and it yields at best a local optimum, this possible heuristic is not pursued. The solutions from the modified K^{th} best algorithm are confirmed globally optimal by the second solution method, the KKT-based MIP reformulation.

Subsequently, the three new NDP models (BLPNDP, SLP2-BLPNDP and RO-BLPNDP) are compared with the existing four NDP models (SONDP, UONDP, SLP2-SONDP and SLP2-UONDP) on Network 1. We find that the BLPNDP yields underestimated solutions (in terms of total system travel time) to the SLP2-BLPNDP in our test problem although this is not always guaranteed. When evaluated all solutions under stochastic condition, SLP2-UONDP and UONDP perform the worst; SONDP and BLPNDP the second worst; SLP2-SONDP the third worst. As expected, SLP2-BLPNDP always yields the least expected TSTT. When comparing between SLP2-BLPNDP, RO-BLPNDP ($\lambda=10$) and RO-BLPNDP ($\lambda=100$), the RO-BLPNDP ($\lambda=100$) yields the least expected risk as it places higher weight on the risk term in the objective function, and the SLP2-BLPNDP yields the highest expected risk. In terms of robustness, we show that RO-BLPNDP yields the most robust solution, and no other model provides a satisfactory solution across the budget range. We also illustrate how a visualized graph may be used to elicit the preference of transportation planners on the desired degree of robustness.

Additionally, we consider the alternative robust NDP model (RO-BLPNDP-VAR) that optimizes the tradeoff between the expected TSTT and the variance of TSTT. RO-BLPNDP-VAR is compared with RO-BLPNDP and SLP2-BLPNDP on Network 1. RO-BLPNDP-VAR yields the least variance of TSTT as it is a component in the objective function. We suggest that RO-BLPNDP-VAR and RO-BLPNDP are appropriate on different settings that highly depend on the preference of transportation planners.

Moreover, we compare the integrated NDP, SSD and DTA models among themselves and against the sequential models on Network 2. With this complexity, we are limited to solving a small network so as to gain insight into the problem and to intuitively understand the effects of alternative improvement policies and signal settings. We find that the BLPNDP-SSD yields underestimated solutions (in terms of total system travel time) to the SLP2-BLPNDP-SSD in our test problem although this is not always guaranteed. From the comparison among the seven integrated models, the RO-BLPNDP-SSD yields the most robust solutions over the budget range, and SLP2-BLPNDP-SSD and SLP2-SONDP-SSD do not always yield the second most robust solution. In SSD, all

models utilize the traffic signal coordination for traffic progression, and the assumption of common cycle length should be relaxed as it constrains the model too severely. From the comparison of the sequential and combined approaches, the integrated robust model (RO-BLPNDP-SSD) performs the best in terms of robustness, and interestingly, the sequential robust approach (RO-BLPNDP & RO-BLPSSD) yields the fifth most robust solution. This reiterates that the sequential robust approach may not be desirable, and the integrated robust approach should be sought after whenever possible.

Additionally, we show two mathematical proofs for reducing a mixed-zero-one continuous linear bi-level program and a mixed-zero-one continuous quadratic-linear bi-level program to a parametric linear bi-level program and a parametric quadratic-linear bi-level program, respectively.

Experimental Results of Metaheuristic Approach

The computational experiences of metaheuristic methods for various transportation network improvement optimization problems are presented using the multi-origin single-destination cell transmission test network (Network 1) and four multi-origin multi-destination test networks (Networks 3.1, 3.2, 4.1 and 4.2). First, the implementation of all metaheuristics and the calibration of SA, MSA and GA parameters for the deterministic NDP are described. Then, the comparison of metaheuristics for the deterministic NDP is discussed. The performance comparison of seven metaheuristics on Network 1 shows that RTS-MT1 outperforms the other algorithms based on the three criteria: solution quality, convergence speed and CPU time. The second best are RTS-MT2 and GA. RTS-MT0 is moderate, and the bottom three algorithms, which perform very poorly, are RS, SA and MSA. For Networks 3.1 and 4.1, only SA, MSA and RS are tested, and the results show that GA appears the best, whereas RS and SA are non-dominated. Also, we empirically show that the metaheuristic parameters are network specific, and they should be calibrated for each network to obtain the best result.

Subsequently, RS, MSA, GA, RTS-MT0, RTS-MT1 and RTS-MT2 are comprehensively compared for the deterministic, stochastic and robust problems of SSD, NDP, NDP-SSD on Networks 3.2 and 4.2. For the stochastic and robust problems, we

use the O-D demand sample sizes of 5 and 3 for Networks 3.2 and 4.2. These O-D demand realizations are independent and identically distributed from the distribution of O-D demands. Since we cannot capture the true CPU time of each metaheuristic run and the functional evaluation time dominates the total CPU time, the CPU times of different metaheuristic algorithms are approximately the same. Also, since the used total trials are relatively small, especially for the stochastic and robust problems, the better solution quality also implies the better convergence speed. As such, we only discuss the comparison in terms of solution quality. The best metaheuristic algorithm is identified for each problem on each network. We find that there is not the single winning metaheuristic across the problems on both networks. For all nine transportation network improvement optimization problems, the rankings in order from best to worst are RTS-MT2, GA, RS, RTS-MT1, MSA and RTS-MT0.

Then, the identified best metaheuristic algorithm for each problem on each network is used to find the near-global optimal solution. For all stochastic and robust problems, we use the same set of 20 O-D demand realizations, which are independent and identically distributed from the distribution of O-D demands. For the robust problems, we set the target TSTT equal to the best expected TSTT from stochastic problems, and use $\lambda = 1$. All deterministic, stochastic and robust SSD and NDP-SSD solutions employ different time offsets and different cycle lengths for intersections on Networks 3.2 and 4.2. This reiterates that we benefit from relaxing the common cycle length assumption typically employed in the literature and from the traffic signal coordination. Subsequently, all metaheuristics solutions are evaluated under stochastic condition by using the new 30 O-D demand realizations in order to eliminate the possible bias. We evaluate the solutions in terms of $E[Robust_Obj(\lambda=1)]$, $E[TSTT]$, $E[Risk]$ and $V[TSTT]$ and also construct the 95% confidence intervals of $E[Robust_Obj(\lambda=1)]$, $E[TSTT]$ and $E[Risk]$.

The comparisons are conducted in three ways on the point estimates. First, when comparing between deterministic, stochastic and robust problems, the robust solutions yield the least $E[Robust_Obj(\lambda=1)]$ and $E[Risk]$, and the deterministic solutions perform

the worst with the greatest $E[Robust_Obj(\lambda=1)]$, $E[Risk]$ and $E[TSTT]$. Since the metaheuristics for the robust problems are not designed to minimize the $V[TSTT]$, the solutions of the robust problems do not always yield the least $V[TSTT]$. Second, when comparing the sequential SSD and NDP approach against the integrated NDP-SSD approach, in terms of $E[Robust_Obj(\lambda=1)]$, the integrated approach performs better than the sequential approach except the deterministic case on Network 4.2. In terms of $E[TSTT]$, the integrated approach performs better than the sequential approach in all cases. In terms of $E[Risk]$, the integrated approach outperforms the sequential approach in all cases except the deterministic case on Network 4.2. Third, when comparing all problems altogether, the RO-NDP-SSD solution outperforms the others on both networks in terms of $E[Robust_Obj(\lambda=1)]$, $E[TSTT]$, $E[Risk]$ and $V[TSTT]$, whereas the sequential robust approach (i.e. RO-SSD & RO-NDP) performs the second best. The solutions from all problems perform better than the original network in terms of $E[Robust_Obj(\lambda=1)]$, $E[TSTT]$ and $E[Risk]$ except certain deterministic cases. This shows that the deterministic approach could yield worse solutions than the do-nothing scenario when evaluating under stochastic condition. Thus, the integrated robust approach (RO-NDP-SSD) should be sought-after whenever possible. Unlike the analytical result in Chapter 5, the sequential robust approach (RO-SSD & RO-NDP) performs the second best. This implies that for larger-size multi-destination networks where the analytical approach is not applicable, the integrated robust approach is the most desirable; otherwise, the next best approach is the sequential robust approach.

Furthermore, we illustrate that a stronger comparison can be made via the paired t test. The 72 paired t tests are conducted to test the null hypotheses that the RO-NDP-SSD solution outperforms the other 12 solutions in $E[TSTT]$, $E[Risk]$ and $E[Robust_Obj(1)]$ on the two networks. The minimal p -value from the 72 paired t tests is 0.794997; thus, we fail to reject the null hypotheses when $\alpha < 0.794997$.

Extensions

We discuss three possible extensions of this dissertation. First, the possible heuristic scheme for BLPNDP is based on the Dantzig-Wolfe decomposition technique.

Second, NDP, SSD and NDP-SSD formulations can be modified to take into account the O-D demand elasticity. Lastly, other promising metaheuristics including hybrid algorithms may be explored to solve our problems.

Appendices

Appendix A. Reduction of MZCL-BLPP to L-BLPP

The proof in this section closely follows the work by Bard (1998) and Kalantari and Rosen (1982). Let's consider the standard optimization problem P1:

$$\begin{aligned} \min_{x_1, x_2} z(x_1, x_2, y) &= c_{11}x_1 + c_{12}x_2 + d_1y \\ \text{subject to} \quad &(x_1, x_2, y) \in P \\ &x_1 \in R_+^{n_1}, \quad x_2 \in \{0,1\}^{n_2}, \quad y \in R_+^m \end{aligned} \quad (\text{P1}) \quad (x_1^*, x_2^*, y^*)$$

where $P = \cup_{i=1}^l P_i'$ such that P_i' is a polyhedral set for $i = 1, 2, \dots, l$ and l is a positive integer. The optimal solution of P1 is denoted by (x_1^*, x_2^*, y^*) . Let's denote the feasible region of P1 by F :

$$F = \{(x_1, x_2, y) \mid (x_1, x_2, y) \in P, x_1 \in R_+^{n_1}, x_2 \in \{0,1\}^{n_2}, y \in R_+^m\}$$

In this section, we will show that the mixed zero-one linear program P1 can be reduced to a linear program P2:

$$\begin{aligned} \min_{x_1, x_2} f(x_1, x_2, y) &= c_{11}x_1 + c_{12}x_2 + d_1y + M\theta(x_2) \\ \text{subject to} \quad &(x_1, x_2, y) \in P \\ &x_1 \in R_+^{n_1}, \quad 0 \leq x_2 \leq e_{n_2}, \quad y \in R_+^m \end{aligned} \quad (\text{P2}) \quad (\tilde{x}_1, \tilde{x}_2, \tilde{y})$$

where $\theta: R^{n_2} \rightarrow R$ is a continuous function such that $\theta(x_2) \geq 0$ for all $0 \leq x_2 \leq e_{n_2}$ and $\theta(x_2) = 0$ if and only if $x_2 \in \{0,1\}^{n_2}$. e_{n_2} is a vector of ones with dimensions n_2 . The optimal solution of P2 is denoted by $(\tilde{x}_1, \tilde{x}_2, \tilde{y})$. We denote the feasible region of P2 by $\tilde{F}$:

$$\tilde{F} = \{(x_1, x_2, y) \mid (x_1, x_2, y) \in P, x_1 \in R_+^{n_1}, 0 \leq x_2 \leq e_{n_2}, y \in R_+^m\}$$

It is noted that $F = (R_+^{n_1} \times \{0,1\}^{n_2} \times R_+^m) \cap \tilde{F}$. It can be assumed without loss of generality that $c_{11} \leq 0$, $c_{12} \leq 0$, and $d_1 \leq 0$. If the i^{th} component of c_{11} (respective c_{12} and d_1) is positive, it is always possible to define a new variable $x'_{1_i} = 1 - x_{1_i}$ ($x'_{2_i} = 1 - x_{2_i}$, $y'_i = 1 - y_i$) and make the appropriate substitutions to yield an equivalent program satisfying this requirement.

Theorem 1: If θ is a concave function, there exists a positive real number M such that the problem P1 and the problem P2 have the same optimal solutions.

Proof

If $\theta(\tilde{x}_2) = 0$, then the result is true for any positive value of M . Let's suppose $\theta(\tilde{x}_2) > 0$. Consider P1 when the integrality requirement is relaxed, giving problem P3:

$$\begin{aligned} \min_{x_1, x_2} z(x_1, x_2, y) &= c_{11}x_1 + c_{12}x_2 + d_1y \\ \text{subject to} \quad (x_1, x_2, y) &\in \tilde{F} \end{aligned} \quad (\text{P3}) \quad (x_1^0, x_2^0, y^0)$$

Let (x_1^0, x_2^0, y^0) denote the optimal solution of P3.

Consider the following P4 with the optimal solution $(\bar{x}_1, \bar{x}_2, \bar{y})$:

$$\begin{aligned} \min \theta(x_2) \\ \text{Subject to} \quad (x_1, x_2, y) &\in \tilde{F} \end{aligned} \quad (\text{P4}) \quad (\bar{x}_1, \bar{x}_2, \bar{y})$$

If $\theta(\bar{x}_2) = 0$, then the result is true for any positive value of M . Suppose that $\theta(\bar{x}_2) > 0$.

Let's define M in P2 such that $M > M_0 = \frac{-c_{11}x_1^0 - c_{12}x_2^0 - d_1y^0}{\theta(\bar{x}_2)}$.

$$\begin{aligned} &f(x_1^*, x_2^*, y^*) - f(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \\ &= c_{11}x_1^* + c_{12}x_2^* + d_1y^* + M\theta(x_2^*) - (c_{11}\tilde{x}_1 + c_{12}\tilde{x}_2 + d_1\tilde{y} + M\theta(\tilde{x}_2)) \\ &= c_{11}x_1^* + c_{12}x_2^* + d_1y^* - (c_{11}\tilde{x}_1 + c_{12}\tilde{x}_2 + d_1\tilde{y} + M\theta(\tilde{x}_2)) \quad (\text{as } x_2^* \in \{0,1\}^{n_2}) \\ &\leq c_{11}x_1^* + c_{12}x_2^* + d_1y^* - (c_{11}\tilde{x}_1 + c_{12}\tilde{x}_2 + d_1\tilde{y} + M\theta(\bar{x}_2)) \quad (\text{as } \theta(\bar{x}_2) \leq \theta(\tilde{x}_2)) \\ &< c_{11}x_1^* + c_{12}x_2^* + d_1y^* - (c_{11}\tilde{x}_1 + c_{12}\tilde{x}_2 + d_1\tilde{y} + M_0\theta(\bar{x}_2)) \quad (\text{as } M > M_0) \\ &= c_{11}x_1^* + c_{12}x_2^* + d_1y^* - (c_{11}\tilde{x}_1 + c_{12}\tilde{x}_2 + d_1\tilde{y} - c_{11}x_1^0 - c_{12}x_2^0 - d_1y^0) \\ &\leq -(c_{11}\tilde{x}_1 + c_{12}\tilde{x}_2 + d_1\tilde{y} - c_{11}x_1^0 - c_{12}x_2^0 - d_1y^0) \\ &(\text{as } c_{11} \leq 0, c_{12} \leq 0, d_1 \leq 0, x_1^* \geq 0, x_2^* \geq 0, y^* \geq 0) \\ &= z(x_1^0, x_2^0, y^0) - z(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \\ &\leq 0 \end{aligned} \quad (\text{from definition of P3})$$

This establishes a contradiction since from the definition of P2,

$$f(x_1^*, x_2^*, y^*) - f(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \geq 0.$$

Thus, $\theta(\bar{x}_2) = 0$, $\theta(\tilde{x}_2) = 0$ and $(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \in F$. From the definition of P2,

$$\begin{aligned} &f(x_1^*, x_2^*, y^*) - f(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \\ &= z(x_1^*, x_2^*, y^*) - z(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \quad (\text{as } \theta(x_2^*) = 0 \text{ and } \theta(\tilde{x}_2) = 0) \\ &\geq 0. \end{aligned}$$

From the definition of P1, $z(x_1^*, x_2^*, y^*) - z(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \leq 0$.

Therefore, $z(x_1^*, x_2^*, y^*) = z(\tilde{x}_1, \tilde{x}_2, \tilde{y})$. □

Theorem 2: There exists a positive real M such that MZCL-BLPP (3.42) and L-BLPP(M) (3.43), have the same optimal solutions.

Proof

From the theoretical properties of L-BLPP in Section 2.7, the inducible region IR can be stated as

$$IR = \{(x_1, x_2, y) : A_{11}x_1 + A_{12}x_2 + B_1y \leq b_1, y \in \arg \min(d_2y : B_2y \leq b_2 - A_{21}x_1 - A_{22}x_2, y \geq 0)\}$$

The inducible region is a finite union of polyhedral sets. Thus, MZCL-BLPP (15.1-15.6) is a special case of problem P1 mentioned in Theorem 1. If we let $\theta(x_2) =$

$$\sum_{j=1}^{n_2} \min\{x_{2j}, 1 - x_{2j}\}, \text{ which is a concave function in } x_2, \text{ then we can apply Theorem 1 to}$$

assure the existence of a real positive M such that MZCL-BLPP and the problem (A.1-A.6) have the same optimal solutions.

$$\min_x F(x, y) = c_{11}x_1 + c_{12}x_2 + d_1y + M \sum_{j=1}^{n_2} \min\{x_{2j}, 1 - x_{2j}\} \quad (\text{A.1})$$

$$\text{subject to} \quad A_{11}x_1 + A_{12}x_2 + B_1y \leq b_1 \quad (\text{A.2})$$

$$x_1 \geq 0, \quad 0 \leq x_2 \leq e_{n_2} \quad (\text{A.3})$$

$$\min_y f(y) = d_2y \quad (\text{A.4})$$

$$\text{subject to} \quad A_{21}x_1 + A_{22}x_2 + B_2y \leq b_2 \quad (\text{A.5})$$

$$y \geq 0 \quad (\text{A.6})$$

However, the problem (A.1-A.6) is equivalent to L-BLPP(M) (3.43.1-3.43.8), since the lower-level optimality in L-BLPP(M) enforces u_j to be equal to $\min\{x_{2j}, 1 - x_{2j}\}$ for all

$j \in \{1, \dots, n_2\}$. We establish the equivalence between MZCL-BLPP and L-BLPP(M). $\square$

The equivalence between (3.42.1-3.42.6) and (3.43.1-3.43.8) is based on the introduction of a concave penalty function and can be used to design penalty methods for solving MZCL-BLPP in the next subsection.

Penalty-Based Algorithm for Solving MZCL-BLPP

Theorem 2 shows that there exists a positive constant M such that the optimal solution of MZCL-BLPP can be obtained by solving a penalized L-BLPP. The value of M is not known in advance. Based on Bard (1998), we suggested solving a series of L-BLPPs for a sequence of M_k values ($k=0, 1, \dots$), starting with

$$M_0 = (c_{11}x_1^0 + c_{12}x_2^0 + d_1y^0) / \theta(x_2^0)$$

where $\theta(x_2^0) = \sum_{j=1}^{n_2} u_j^0$; and (x_1^0, x_2^0, y^0) is an initial point, and can be taken as the

optimum of the relaxed version of MZCL-BLPP. The sequence terminates when a value M' has been found such that the optimal solution of L-BLPP(M') is integral with respect to the upper-level zero-one variables. The corresponding optimal solution of this L-BLPP(M') is an optimal solution of MZCL-BLPP. Each L-BLPP(M) can be solved by KKT-based MIP reformulation as discussed in Section 3.4.1.2.

Appendix B. Reduction of MZC-QL-BLPP to QL-BLPP

The proof in this section is similar to Appendix A, and closely follows the work by Bard (1998) and Kalantari and Rosen (1989). Let's consider the standard optimization problem P5:

$$\begin{aligned} \min_{x_1, x_2} z(x_1, x_2, y) &= c_{11}x_1 + c_{12}x_2 + d_1y + \frac{1}{2}y^T Qy \\ \text{subject to} \quad &(x_1, x_2, y) \in P \\ &x_1 \in R_+^{n_1}, \quad x_2 \in \{0,1\}^{n_2}, \quad y \in R_+^m \end{aligned} \quad (\text{P5}) \quad (x_1^*, x_2^*, y^*)$$

where $P = \bigcup_{i=1}^l P'_i$ such that P'_i is a polyhedral set for $i = 1, 2, \dots, l$ and l is a positive integer. The optimal solution of P5 is denoted by (x_1^*, x_2^*, y^*) . Let's denote the feasible region of P5 as F :

$$F = \{(x_1, x_2, y) \mid (x_1, x_2, y) \in P, x_1 \in R_+^{n_1}, x_2 \in \{0,1\}^{n_2}, y \in R_+^m\}$$

In this section, we will show that the mixed zero-one quadratic program P5 can be reduced to a quadratic program P6:

$$\begin{aligned} \min_{x_1, x_2} f(x_1, x_2, y) &= c_{11}x_1 + c_{12}x_2 + d_1y + \frac{1}{2}y^T Qy + M\theta(x_2) \\ \text{subject to} \quad &(x_1, x_2, y) \in P \\ &x_1 \in R_+^{n_1}, \quad 0 \leq x_2 \leq e_{n_2}, \quad y \in R_+^m \end{aligned} \quad (\text{P6}) \quad (\tilde{x}_1, \tilde{x}_2, \tilde{y})$$

where $\theta: R^{n_2} \rightarrow R$ is a continuous function such that $\theta(x_2) \geq 0$ for all $0 \leq x_2 \leq e_{n_2}$ and $\theta(x_2) = 0$ if and only if $x_2 \in \{0,1\}^{n_2}$. e_{n_2} is a vector of ones with dimensions n_2 . The optimal solution of P6 is denoted by $(\tilde{x}_1, \tilde{x}_2, \tilde{y})$. We denote the feasible region of P6 as $\tilde{F}$:

$$\tilde{F} = \{(x_1, x_2, y) \mid (x_1, x_2, y) \in P, x_1 \in R_+^{n_1}, 0 \leq x_2 \leq e_{n_2}, y \in R_+^m\}$$

It is noted that $F = (R_+^{n_1} \times \{0,1\}^{n_2} \times R_+^m) \cap \tilde{F}$. It can be assumed without loss of generality that $c_{11} \leq 0$, $c_{12} \leq 0$, and $d_1 \leq 0$. If the i^{th} component of c_{11} (respective c_{12} and d_1) is positive, it is always possible to define a new variable $x'_{1_i} = 1 - x_{1_i}$ ($x'_{2_i} = 1 - x_{2_i}$, $y'_i = 1 - y_i$) and make the appropriate substitutions to yield an equivalent program satisfying this requirement.

Theorem 3: If θ is a concave function, there exists a positive real number M such that the problem P5 and the problem P6 have the same optimal solutions.

Proof

If $\theta(\tilde{x}_2) = 0$, then the result is true for any positive value of M . Let's suppose $\theta(\tilde{x}_2) > 0$. Consider P5 when the integrality requirement is relaxed, giving problem P7:

$$\begin{aligned} \min_{x_1, x_2} z(x_1, x_2, y) &= c_{11}x_1 + c_{12}x_2 + d_1y + \frac{1}{2}y^T Qy \\ \text{subject to} \quad &(x_1, x_2, y) \in \tilde{F} \end{aligned} \quad (\text{P7}) \quad (x_1^0, x_2^0, y^0)$$

Let (x_1^0, x_2^0, y^0) denote the optimal solution of P7.

Consider the following P8 with the optimal solution $(\bar{x}_1, \bar{x}_2, \bar{y})$:

$$\begin{aligned} \min \theta(x_2) \\ \text{subject to} \quad &(x_1, x_2, y) \in \tilde{F} \end{aligned} \quad (\text{P8}) \quad ((\bar{x}_1, \bar{x}_2, \bar{y}))$$

If $\theta(\bar{x}_2) = 0$, then the result is true for any positive value of M . Suppose that $\theta(\bar{x}_2) > 0$. Let's define M in P6 such that

$$M > M_0 = \frac{-c_{11}x_1^0 - c_{12}x_2^0 - d_1y^0 - \frac{1}{2}y^{0T}Qy^0 + \frac{1}{2}y^{*T}Qy^*}{\theta(\bar{x}_2)} = \frac{-z(x_1^0, x_2^0, y^0) + \frac{1}{2}y^{*T}Qy^*}{\theta(\bar{x}_2)}$$

$$\begin{aligned} &f(x_1^*, x_2^*, y^*) - f(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \\ &= c_{11}x_1^* + c_{12}x_2^* + d_1y^* + \frac{1}{2}y^{*T}Qy^* + M\theta(x_2^*) - \left(c_{11}\tilde{x}_1 + c_{12}\tilde{x}_2 + d_1\tilde{y} + \frac{1}{2}\tilde{y}^TQ\tilde{y} + M\theta(\tilde{x}_2) \right) \\ &= c_{11}x_1^* + c_{12}x_2^* + d_1y^* + \frac{1}{2}y^{*T}Qy^* - \left(c_{11}\tilde{x}_1 + c_{12}\tilde{x}_2 + d_1\tilde{y} + \frac{1}{2}\tilde{y}^TQ\tilde{y} + M\theta(\tilde{x}_2) \right) \\ &(\text{as } x_2^* \in \{0,1\}^{n_2}) \\ &\leq c_{11}x_1^* + c_{12}x_2^* + d_1y^* + \frac{1}{2}y^{*T}Qy^* - \left(c_{11}\tilde{x}_1 + c_{12}\tilde{x}_2 + d_1\tilde{y} + \frac{1}{2}\tilde{y}^TQ\tilde{y} + M\theta(\bar{x}_2) \right) \\ &(\text{as } \theta(\bar{x}_2) \leq \theta(\tilde{x}_2)) \\ &< c_{11}x_1^* + c_{12}x_2^* + d_1y^* + \frac{1}{2}y^{*T}Qy^* - \left(c_{11}\tilde{x}_1 + c_{12}\tilde{x}_2 + d_1\tilde{y} + \frac{1}{2}\tilde{y}^TQ\tilde{y} + M_0\theta(\bar{x}_2) \right) \\ &(\text{as } M > M_0) \\ &\leq \frac{1}{2}y^{*T}Qy^* - \left(c_{11}\tilde{x}_1 + c_{12}\tilde{x}_2 + d_1\tilde{y} + \frac{1}{2}\tilde{y}^TQ\tilde{y} + M_0\theta(\bar{x}_2) \right) \\ &(\text{as } c_{11}, c_{12}, d_1 \leq 0 \text{ and } x_1^*, x_2^*, y^* \geq 0) \\ &= \frac{1}{2}y^{*T}Qy^* - (z(\tilde{x}_1, \tilde{x}_2, \tilde{y}) + M_0\theta(\bar{x}_2)) \end{aligned}$$

$$\begin{aligned}
&= \frac{1}{2} y^{*T} Q y^* - \left(z(\tilde{x}_1, \tilde{x}_2, \tilde{y}) - z(x_1^0, x_2^0, y^0) + \frac{1}{2} y^{*T} Q y^* \right) \\
&\quad \left(\text{as } M_0 = \frac{-z(x_1^0, x_2^0, y^0) + \frac{1}{2} y^{*T} Q y^*}{\theta(\bar{x}_2)} \right) \\
&= z(x_1^0, x_2^0, y^0) - z(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \\
&\leq 0 \quad \text{(from definition of P7)}
\end{aligned}$$

This establishes a contradiction since from the definition of P6,

$$f(x_1^*, x_2^*, y^*) - f(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \geq 0.$$

Thus, $\theta(\bar{x}_2) = 0$, $\theta(\tilde{x}_2) = 0$ and $(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \in F$. From the definition of P6,

$$\begin{aligned}
&f(x_1^*, x_2^*, y^*) - f(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \\
&= z(x_1^*, x_2^*, y^*) - z(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \quad \text{(as } \theta(x_2^*) = 0 \text{ and } \theta(\tilde{x}_2) = 0) \\
&\geq 0.
\end{aligned}$$

From the definition of P5, $z(x_1^*, x_2^*, y^*) - z(\tilde{x}_1, \tilde{x}_2, \tilde{y}) \leq 0$.

Therefore, $z(x_1^*, x_2^*, y^*) = z(\tilde{x}_1, \tilde{x}_2, \tilde{y})$. □

Theorem 4: There exists a positive real M such that MZC-QL-BLPP (3.45.1-3.45.6) and QL-BLPP(M) (3.46.1-3.46.8), have the same optimal solutions.

Proof

From the theoretical properties of L-BLPP in Bard (1998), the inducible region IR can be stated as

$$IR = \{(x_1, x_2, y) : A_{11}x_1 + A_{12}x_2 + B_1y \leq b_1, y \in \arg \min(d_2y : B_2y \leq b_2 - A_{21}x_1 - A_{22}x_2, y \geq 0)\}.$$

The inducible region is a finite union of polyhedral sets. Thus, MZC-QL-BLPP is a special case of problem P5 mentioned in Theorem 3. If we define

$$\theta(x_2) = \sum_{j=1}^{n_2} \min\{x_{2j}, 1 - x_{2j}\}, \text{ which is a concave function in } x_2, \text{ then we can apply}$$

Theorem 3 to assure the existence of a real positive M such that MZC-QL-BLPP and the problem (B.1-B.6) have the same optimal solutions.

$$\min_x F(x, y) = c_{11}x_1 + c_{12}x_2 + d_1y + \frac{1}{2} y^T Q y + M \sum_{j=1}^{n_2} \min\{x_{2j}, 1 - x_{2j}\} \quad (B.1)$$

$$\text{subject to } A_{11}x_1 + A_{12}x_2 + B_1y \leq b_1 \quad (B.2)$$

$$x_1 \geq 0, \quad 0 \leq x_2 \leq e_{n_2} \quad (B.3)$$

$$\min_y f(y) = d_2y \quad (B.4)$$

$$\text{subject to } A_{21}x_1 + A_{22}x_2 + B_2y \leq b_2 \quad (B.5)$$

$$y \geq 0 \tag{B.6}$$

However, the problem (B.1-B.6) is equivalent to QL-BLPP(M) (3.46.1-3.46.8), since the lower-level optimality in QL-BLPP(M) enforces u_j to be equal to $\min\{x_{2j}, 1 - x_{2j}\}$ for all $j \in \{1, \dots, n_2\}$. We establish the equivalence between MZC-QL-BLPP and QL-BLPP(M).

□

The equivalence between (3.45.1-3.45.6) and (3.46.1-3.46.8) is based on the introduction of a concave penalty function and can be used to design penalty methods for solving MZC-QL-BLPP in the next section.

Penalty-Based Algorithm for Solving MZC-QL-BLPP

Theorem 4 shows that there exists a positive constant M such that the optimal solution of MZC-QL-BLPP can be obtained by solving a penalized QL-BLPP. The value of M is not known in advance. Based on Bard (1998), we suggested solving a series of QL-BLPPs for a sequence of M_k values ($k=0, 1, \dots$), starting with

$$M_0 = \left(c_{11}x_1^0 + c_{12}x_2^0 + d_1y^0 + \frac{1}{2}y^{0T}Qy^0 \right) / \theta(x_2^0)$$

where $\theta(x_2^0) = \sum_{j=1}^{n_2} u_j^0$; and (x_1^0, x_2^0, y^0) is an initial point, and can be taken as the

optimum of the relaxed version of MZC-QL-BLPP. The sequence terminates when a value M' has been found such that the optimal solution of QL-BLPP(M') is integral with respect to the upper-level zero-one variables. The corresponding optimal solution of this QL-BLPP is an optimal solution of MZC-QL-BLPP. Then, QL-BLPP(M) can be solved by KKT-based MINLP reformulation. If the quadratic objective function is convex, the global optimal solution can be found by any MINLP solver (GAMS/DICOPT in this study).

Appendix C. Detailed Solutions of NDP Models in Section 5.1.3

Table C.1. Solutions of SONDP at Different Total Budget Levels

SONDP	Total Available Budget									
	37	38	39	40	41	42	43	44	45	46
b1	11.375	11.75	12.125	12.5	12.875	13.25	13.625	14	14.375	14.75
b2	15	15	15	15	15	15	15	15	15	15
b3	0	0	0	0	0	0	0	0	0	0
b4	0	0	0	0	0	0	0	0	0	0
b5	10.625	11.25	11.875	12.5	13.125	13.75	14.375	15	15.625	16.25
Obj. Value	12.362	12.325	12.287	12.25	12.212	12.175	12.137	12.1	12.062	12.025
$E[UO_TSTT]$	13.544	13.488	13.431	13.375	13.338	13.3	13.263	13.225	13.188	13.15
$TSTT^{TARGET}$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
TSTT1	13.225	13.15	13.075	13	12.962	12.925	12.887	12.85	12.812	12.775
$(Risk1)^{0.5}$	0	0	0	0	0	0	0	0	0	0
TSTT2	13.862	13.825	13.787	13.75	13.712	13.675	13.637	13.6	13.562	13.525
$(Risk2)^{0.5}$	0.387	0.425	0.462	0.5	0.537	0.575	0.612	0.65	0.687	0.725
$E[Risk]$	0.075	0.09	0.107	0.125	0.144	0.165	0.188	0.211	0.236	0.263

Table C.2. Solutions of UONDP at Different Total Budget Levels

SONDP	Total Available Budget									
	37	38	39	40	41	42	43	44	45	46
b1	3.333	3.333	3.333	5	5.375	5.75	6.125	6.5	6.875	7.25
b2	10.542	10.917	11.292	10	10	10	10	10	10	10
b3	0	0	0	0	0	0	0	0	0	0
b4	0	0	0	0	0	0	0	0	0	0
b5	23.125	23.75	24.375	25	25.625	26.25	26.875	27.5	28.125	28.75
Obj. Value	12.612	12.575	12.537	12.5	12.462	12.425	12.387	12.35	12.312	12.275
$E[UO_TSTT]$	14.072	14.006	13.941	13.625	13.559	13.494	13.428	13.363	13.297	13.231
$TSTT^{TARGET}$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
TSTT1	14.473	14.454	14.436	13.75	13.656	13.562	13.469	13.375	13.281	13.187
$(Risk1)^{0.5}$	0.998	1.054	1.111	0.5	0.481	0.462	0.444	0.425	0.406	0.387
TSTT2	13.671	13.558	13.446	13.5	13.462	13.425	13.387	13.35	13.312	13.275
$(Risk2)^{0.5}$	0.196	0.158	0.121	0.25	0.287	0.325	0.362	0.4	0.437	0.475
$E[Risk]$	0.517	0.568	0.624	0.156	0.157	0.16	0.164	0.17	0.178	0.188

Table C.3. Solutions of BLPNDP at Different Total Budget Levels

BLPNDP	Total Available Budget									
	37	38	39	40	41	42	43	44	45	46
b1	11.375	11.75	12.125	12.5	12.875	13.25	13.625	14	14.375	14.75
b2	15	15	15	15	15	15	15	15	15	15
b3	0	0	0	0	0	0	0	0	0	0
b4	0	0	0	0	0	0	0	0	0	0
b5	10.625	11.25	11.875	12.5	13.125	13.75	14.375	15	15.625	16.25
Obj. Value	12.362	12.325	12.287	12.25	12.212	12.175	12.137	12.1	12.062	12.025
$E[UO_TSTT]$	13.544	13.488	13.431	13.375	13.338	13.3	13.263	13.225	13.188	13.15
$TSTT^{TARGET}$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
TSTT1	13.225	13.15	13.075	13	12.962	12.925	12.887	12.85	12.812	12.775
$(Risk1)^{0.5}$	0	0	0	0	0	0	0	0	0	0
TSTT2	13.862	13.825	13.787	13.75	13.712	13.675	13.637	13.6	13.562	13.525
$(Risk2)^{0.5}$	0.387	0.425	0.462	0.5	0.537	0.575	0.612	0.65	0.687	0.725
$E[Risk]$	0.075	0.09	0.107	0.125	0.144	0.165	0.188	0.211	0.236	0.263

Table C.4. Solutions of SLP2-SONDP at Different Total Budget Levels

SLP2-SONDP	Total Available Budget									
	37	38	39	40	41	42	43	44	45	46
b1	10	10	10	10	10	10	10	10	10	10
b2	10.333	11.333	12.333	13.333	14.333	15.333	16.333	17.333	18.333	19.333
b3	0	0	0	0	0	0	0	0	0	0
b4	0	0	0	0	0	0	0	0	0	0
b5	16.667	16.667	16.667	16.667	16.667	16.667	16.667	16.667	16.667	16.667
Obj. Value	13.717	13.617	13.517	13.417	13.317	13.217	13.117	13.017	12.917	12.817
$E[UO_TSTT]$	13.717	13.617	13.517	13.417	13.317	13.217	13.117	13.017	12.917	12.817
$TSTT^{TARGET}$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
TSTT1	13	13	13	13	13	13	13	13	13	13
$(Risk1)^{0.5}$	0	0	0	0	0	0	0	0.05	0.125	0.2
TSTT2	14.433	14.233	14.033	13.833	13.633	13.433	13.233	13.033	12.833	12.633
$(Risk2)^{0.5}$	0.958	0.833	0.708	0.583	0.458	0.333	0.208	0.083	0	0
$E[Risk]$	0.459	0.347	0.251	0.17	0.105	0.056	0.022	0.005	0.008	0.02

Table C.5. Solutions of SLP2-UONDP at Different Total Budget Levels

SLP2-UONDP	Total Available Budget									
	37	38	39	40	41	42	43	44	45	46
b1	10.091	10.363	10.636	10.909	11.182	11.455	11.727	12	12.273	12.546
b2	10.091	10.363	10.636	10.909	11.182	11.455	11.727	12	12.273	12.546
b3	0	0	0	0	0	0	0	0	0	0
b4	0	0	0	0	0	0	0	0	0	0
b5	16.818	17.274	17.728	18.182	18.636	19.090	19.546	20	20.454	20.908
Obj. Value	13.732	13.677	13.622	13.568	13.514	13.459	13.405	13.35	13.296	13.241
$E[UO_TSTT]$	13.732	13.677	13.622	13.568	13.514	13.459	13.405	13.35	13.296	13.241
$TSTT^{TARGET}$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
TSTT1	12.991	12.964	12.936	12.909	12.882	12.855	12.827	12.8	12.773	12.745
$(Risk1)^{0.5}$	0	0	0	0	0	0	0	0	0	0
TSTT2	14.473	14.391	14.309	14.227	14.145	14.064	13.982	13.9	13.818	13.736
$(Risk2)^{0.5}$	0.998	0.991	0.984	0.977	0.97	0.964	0.957	0.95	0.943	0.936
$E[Risk]$	0.498	0.491	0.484	0.478	0.471	0.464	0.458	0.451	0.445	0.438

Table C.6. Solutions of SLP2-BLPNDP at Different Total Budget Levels

SLP2-BLPNDP	Total Available Budget									
	37	38	39	40	41	42	43	44	45	46
b1	10	10	10	10	10	10	10	10	10	10
b2	16.375	16.75	17.125	17.5	17.875	18.25	18.625	19	19.375	19.75
b3	0	0	0	0	0	0	0	0	0	0
b4	0	0	0	0	0	0	0	0	0	0
b5	10.625	11.25	11.875	12.5	13.125	13.75	14.375	15	15.625	16.25
Obj. Value	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
$E[UO_TSTT]$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
$TSTT^{TARGET}$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
TSTT1	13.362	13.325	13.287	13.25	13.212	13.165	13.137	13.1	13.062	13.025
$(Risk1)^{0.5}$	0	0	0	0	0.037	0.065	0.112	0.15	0.187	0.225
TSTT2	13.587	13.475	13.362	13.25	13.137	13.035	12.912	12.8	12.687	12.575
$(Risk2)^{0.5}$	0.112	0.075	0.037	0	0	0	0	0	0	0
$E[Risk]$	0.006	0.003	0.0007 031	0	0.0007 03	0.0021 12	0.006	0.011	0.018	0.025

Table C.7. Solutions of RO-BLPNDP ($\lambda = 10$) at Different Total Budget Levels

RO-BLPNDP ($\lambda = 10$)	Total Available Budget									
	37	38	39	40	41	42	43	44	45	46
b1	9.593	10	10	10	10	10	11.038	11.434	11	12.046
b2	16.528	16.75	17.125	17.5	17.875	18.25	18.236	18.462	19	18.983
b3	0	0	0	0	0	0	0	0	0	0
b4	0	0	0	0	0	0	0	0	0	0
b5	10.879	11.25	11.875	12.5	13.125	13.75	13.726	14.104	15	14.971
Obj. Value	13.528	13.428	13.332	13.25	13.182	13.121	13.062	13.005	12.978	12.987
$E[UO_TSTT]$	13.506	13.4	13.325	13.25	13.175	13.1	13.051	12.986	12.9	12.877
$TSTT^{TARGET}$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
TSTT1	13.469	13.325	13.287	13.25	13.212	13.165	13.073	13.01	13	12.949
$(Risk1)^{0.5}$	0	0	0	0	0.037	0.065	0.048	0.06	0.125	0.149
TSTT2	13.542	13.475	13.362	13.25	13.137	13.035	13.029	12.961	12.8	12.805
$(Risk2)^{0.5}$	0.067	0.075	0.037	0	0	0	0.004	0.011	0	0.005
$E[Risk]$	0.0022 4	0.003	0.0007 031	0	0.0007 03	0.0021 12	0.001	0.0019 2	0.008	0.011

Table C.8. Solutions of RO-BLPNDP ($\lambda = 100$) at Different Total Budget Levels

RO-BLPNDP ($\lambda = 100$)	Total Available Budget									
	37	38	39	40	41	42	43	44	45	46
b1	9.492	9.664	9.837	10	10.381	10.777	11.173	11.570	11.876	12.314
b2	16.565	16.876	17.186	17.5	17.732	17.959	18.185	18.411	18.672	18.882
b3	0	0	0	0	0	0	0	0	0	0
b4	0	0	0	0	0	0	0	0	0	0
b5	10.943	11.460	11.977	12.5	12.887	13.264	13.642	14.019	14.452	14.804
Obj. Value	13.688	13.503	13.357	13.25	13.195	13.162	13.15	13.159	13.44	13.933
$E[UO_TSTT]$	13.513	13.426	13.337	13.25	13.185	13.119	13.054	12.989	12.936	12.888
$TSTT^{TARGET}$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
TSTT1	13.496	13.413	13.33	13.25	13.189	13.126	13.064	13.002	12.973	12.94
$(Risk1)^{0.5}$	0.021	0.013	0.005	0	0.014	0.026	0.039	0.052	0.098	0.14
TSTT2	13.53	13.437	13.344	13.25	13.18	13.112	13.045	12.977	12.899	12.835
$(Risk2)^{0.5}$	0.055	0.037	0.019	0	0.005	0.012	0.02	0.027	0.024	0.035
$E[Risk]$	0.0017 5	0.0007 79	0.0001 97	0	0.0001 08	0.0004 26	0.0009 57	0.0016 98	0.005	0.01

Appendix D. Detailed Solutions and Robustness Evaluation Results in Section 5.1.4

Table D.1. Solutions of SLP2-BLPNDP and Robustness Evaluations Based on Fixed and Moving Targets at Different Total Budget Levels

SLP2-BLPNDP	37	38	39	40	41	42	43	44	45	46
b1	10	10	10	10	10	10	10	10	10	10
b2	16.375	16.75	17.125	17.5	17.875	18.25	18.625	19	19.375	19.75
b3	0	0	0	0	0	0	0	0	0	0
b4	0	0	0	0	0	0	0	0	0	0
b5	10.625	11.25	11.875	12.5	13.125	13.75	14.375	15	15.625	16.25
$E[UO_TSTT]$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
$V[UO_TSTT]$	1.27E-02	5.63E-03	1.41E-03	0.00E+00	1.41E-03	5.63E-03	1.27E-02	2.25E-02	3.52E-02	5.06E-02
$TSTT^{TARGET}$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
TSTT1	13.362	13.325	13.287	13.25	13.212	13.175	13.137	13.1	13.062	13.025
TSTT2	13.587	13.475	13.362	13.25	13.137	13.025	12.912	12.8	12.687	12.575
$E[Risk]$	6.27E-03	2.81E-03	6.85E-04	0.00E+00	6.84E-04	2.81E-03	6.27E-03	1.13E-02	1.75E-02	2.53E-02
$E[Robust_Obj(10)]$	13.538	13.428	13.332	13.250	13.182	13.128	13.088	13.063	13.050	13.053
$E[Robust_Obj(100)]$	14.102	13.681	13.393	13.250	13.243	13.381	13.652	14.075	14.623	15.331
$E[Robust_Obj_Var(10)]$	13.602	13.456	13.339	13.250	13.189	13.156	13.152	13.175	13.227	13.306
$E[Robust_Obj_Var(100)]$	14.741	13.963	13.466	13.250	13.316	13.663	14.291	15.200	16.391	17.863

Table D.2. Solutions of RO-BLPNDP($\lambda = 10$) and Robustness Evaluations Based on Fixed and Moving Targets at Different Total Budget Levels

RO-BLPNDP ($\lambda = 10$)	37	38	39	40	41	42	43	44	45	46
b1	9.593	9.926	10.000	10.000	10.000	10.560	11.038	11.434	11.000	12.046
b2	16.528	16.778	17.125	17.500	17.875	18.040	18.236	18.462	19.000	18.983
b3	0	0	0	0	0	0	0	0	0	0
b4	0	0	0	0	0	0	0	0	0	0
b5	10.880	11.296	11.875	12.500	13.125	13.400	13.726	14.104	15.000	14.971
$E[UO_TSTT]$	13.506	13.406	13.325	13.250	13.175	13.114	13.051	12.986	12.900	12.877
$V[UO_TSTT]$	1.33E-03	3.78E-03	1.41E-03	0.00E+00	1.41E-03	6.76E-04	4.84E-04	6.00E-04	1.00E-02	5.18E-03
$TSTT^{TARGET}$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
TSTT1	13.469	13.344	13.287	13.25	13.212	13.14	13.073	13.01	13	12.949
TSTT2	13.542	13.467	13.362	13.25	13.137	13.088	13.029	12.961	12.8	12.805
$E[Risk]$	2.24E-03	2.24E-03	6.85E-04	0.00E+00	6.84E-04	8.00E-04	1.16E-03	1.86E-03	7.81E-03	1.11E-02
$E[Robust_Obj(10)]$	13.528	13.428	13.332	13.250	13.182	13.122	13.063	13.004	12.978	12.988
$E[Robust_Obj(100)]$	13.730	13.630	13.393	13.250	13.243	13.194	13.167	13.172	13.681	13.988
$E[Robust_Obj_Var(10)]$	13.519	13.443	13.339	13.250	13.189	13.121	13.056	12.992	13.000	12.929
$E[Robust_Obj_Var(100)]$	13.639	13.784	13.466	13.250	13.316	13.182	13.099	13.046	13.900	13.395

Table D.3. Solutions of RO-BLPNDP($\lambda = 100$) and Robustness Evaluations Based on Fixed and Moving Targets at Different Total Budget Levels

RO-BLPNDP ($\lambda = 100$)	37	38	39	40	41	42	43	44	45	46
b1	9.492	9.664	9.837	10.000	10.381	10.777	11.174	11.570	11.876	12.314
b2	16.566	16.876	17.186	17.500	17.732	17.959	18.185	18.411	18.672	18.882
b3	0	0	0	0	0	0	0	0	0	0
b4	0	0	0	0	0	0	0	0	0	0
b5	10.943	11.460	11.977	12.500	12.887	13.264	13.642	14.019	14.453	14.804
$E[UO_TSTT]$	13.513	13.425	13.337	13.250	13.185	13.119	13.054	12.989	12.936	12.888
$V[UO_TSTT]$	2.89E-04	1.44E-04	4.90E-05	0.00E+00	2.03E-05	4.92E-05	9.03E-05	1.56E-04	1.37E-03	2.76E-03
$TSTT^{TARGET}$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
TSTT1	13.496	13.413	13.33	13.25	13.189	13.126	13.064	13.002	12.973	12.94
TSTT2	13.53	13.437	13.344	13.25	13.18	13.112	13.045	12.977	12.899	12.835
$E[Risk]$	1.73E-03	7.69E-04	1.93E-04	0.00E+00	1.10E-04	4.10E-04	9.60E-04	1.72E-03	5.09E-03	1.04E-02
$E[Robust_Obj(10)]$	13.530	13.433	13.339	13.250	13.186	13.124	13.064	13.006	12.987	12.992
$E[Robust_Obj(100)]$	13.686	13.502	13.357	13.250	13.196	13.160	13.150	13.161	13.445	13.929
$E[Robust_Obj_Var(10)]$	13.516	13.427	13.338	13.250	13.185	13.120	13.055	12.991	12.949	12.915
$E[Robust_Obj_Var(100)]$	13.542	13.440	13.342	13.250	13.187	13.124	13.063	13.005	13.073	13.163

Table D.4. Solutions of RO-BLPNDP-VAR($\lambda = 10$) and Robustness Evaluations Based on Fixed and Moving Targets at Different Total Budget Levels

RO-BLPNDP-VAR ($\lambda = 10$)	37	38	39	40	41	42	43	44	45	46
b1	9.507	9.707	9.907	10.000	10.265	10.694	11.122	11.551	11.998	12.650
b2	16.560	16.860	17.160	17.500	17.776	17.990	18.204	18.418	18.626	18.756
b3	0	0	0	0	0	0	0	0	0	0
b4	0	0	0	0	0	0	0	0	0	0
b5	10.933	11.433	11.933	12.500	12.959	13.316	13.674	14.031	14.376	14.594
$E[UO_TSTT]$	13.512	13.422	13.332	13.250	13.182	13.117	13.053	12.989	12.941	12.901
$V[UO_TSTT]$	4.00E-04	4.00E-04	4.00E-04	0.00E+00	2.10E-04	2.10E-04	1.96E-04	2.10E-04	8.12E-04	8.12E-04
$TSTT^{TARGET}$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
TSTT1	13.492	13.402	13.312	13.25	13.196	13.132	13.067	13.003	12.969	12.93
TSTT2	13.532	13.442	13.352	13.25	13.167	13.103	13.039	12.974	12.912	12.873
$E[Risk]$	1.77E-03	8.84E-04	3.65E-04	0.00E+00	2.20E-04	5.17E-04	9.80E-04	1.69E-03	5.10E-03	1.11E-02
$E[Robust_Obj(10)]$	13.530	13.431	13.336	13.250	13.184	13.122	13.063	13.006	12.992	13.013
$E[Robust_Obj(100)]$	13.689	13.510	13.368	13.250	13.204	13.169	13.151	13.158	13.451	14.013
$E[Robust_Obj_Var(10)]$	13.516	13.426	13.336	13.250	13.184	13.119	13.055	12.991	12.949	12.910
$E[Robust_Obj_Var(100)]$	13.552	13.462	13.372	13.250	13.203	13.138	13.073	13.010	13.022	12.983

Table D.5. Solutions of RO-BLPNDP-VAR($\lambda=100$) and Robustness Evaluations Based on Fixed and Moving Targets at Different Total Budget Levels

RO-BLPNDP-VAR ($\lambda=100$)	37	38	39	40	41	42	43	44	45	46
b1	9.411	9.611	9.811	10.000	10.412	10.841	11.269	11.700	12.352	13.004
b2	16.596	16.896	17.196	17.500	17.720	17.935	18.149	18.363	18.493	18.623
b3	0	0	0	0	0	0	0	0	0	0
b4	0	0	0	0	0	0	0	0	0	0
b5	10.993	11.493	11.993	12.500	12.867	13.225	13.582	13.938	14.155	14.372
$E[UO_TSTT]$	13.519	13.429	13.339	13.250	13.185	13.121	13.057	12.994	12.955	12.916
$V[UO_TSTT]$	4.04E-06	4.04E-06	4.04E-06	0	2.29E-06	1E-06	2.29E-06	9.01E-06	9.01E-06	9.04E-06
$TSTT^{TARGET}$	13.475	13.4	13.325	13.25	13.175	13.1	13.025	12.95	12.875	12.8
TSTT1	13.517	13.427	13.337	13.25	13.187	13.122	13.058	12.997	12.958	12.919
TSTT2	13.521	13.431	13.341	13.25	13.184	13.12	13.055	12.991	12.952	12.913
$E[Risk]$	1.94E-03	8.45E-04	2.00E-04	0.00E+00	1.12E-04	4.42E-04	9.94E-04	1.95E-03	6.41E-03	1.35E-02
$E[Robust_Obj(10)]$	13.539	13.438	13.341	13.250	13.186	13.125	13.067	13.014	13.019	13.050
$E[Robust_Obj(100)]$	13.713	13.514	13.359	13.250	13.197	13.165	13.156	13.189	13.596	14.262
$E[Robust_Obj_Var(10)]$	13.519	13.429	13.339	13.250	13.185	13.121	13.057	12.994	12.955	12.916
$E[Robust_Obj_Var(100)]$	13.520	13.430	13.340	13.250	13.186	13.121	13.057	12.995	12.956	12.917

Appendix E. Detailed Solutions of Combined NDP, SSD and DTA Models in Section 5.2.2

Table E.1. Solutions of SONDP-SSD at Different Total Budget Levels

SONDP-SSD	Total Available Budget									
	6	7	8	9	10	11	12	13	14	15
b1	1.6	2.2	2.8	3.4	4	4.6	0	0.5	1.5	2.5
b2	0	0	0	0	0	0	5	5	5	5
b3	4.4	4.8	5.2	5.6	6	6.4	7	7.5	7.5	7.5
b4	0	0	0	0	0	0	0	0	0	0
b5	0	0	0	0	0	0	0	0	0	0
b6	0	0	0	0	0	0	0	0	0	0
c3	s9	s8	s8	s8	s8	s8	s5	s5	s5	s5
c5	s1008	s1008	s1008	s1008	s1008	s1008	s1005	s1005	s1005	s1005
c6	s2006	s2006	s2010	s2006	s2006	s2006	s2006	s2006	s2006	s2006
Obj. Value	14.52	14.34	14.16	13.98	13.8	13.62	13.45	13.35	13.3	13.25
$E[UO_TSTT]$	15.1	14.81	14.980	14.57	14.450	14.330	14.200	14.087	14.012	13.937
$TSTT^{TARGET}$	14.88 3	14.76 7	14.65	14.53 3	14.41 7	14.3	14.18 3	14.06 7	13.95	13.83 3
TSTT1	20.86 0	20.34 0	20.74 0	19.98 0	19.80 0	19.62 0	19.45 0	19.32 5	19.22 5	19.12 5
$(Risk1)^{0.5}$	5.977	5.573	6.090	5.447	5.383	5.320	5.267	5.258	5.275	5.292
TSTT2	9.340	9.280	9.220	9.160	9.100	9.040	8.950	8.850	8.800	8.750
$(Risk2)^{0.5}$	0	0	0	0	0	0	0	0	0	0
$E[Risk]$	17.862	15.529	18.544	14.835	14.488	14.151	13.871	13.823	13.913	14.003

Table E.2. Solutions of UONDP-SSD at Different Total Budget Levels

UONDP-SSD	Total Available Budget									
	6	7	8	9	10	11	12	13	14	15
b1	0	0	0	0	0	1	0	3	0	0
b2	0	0	0	0	0	0	2	0	4	5
b3	6	7	8	9	10	10	10	10	10	10
b4	0	0	0	0	0	0	0	0	0	0
b5	0	0	0	0	0	0	0	0	0	0
b6	0	0	0	0	0	0	0	0	0	0
c3	s1	s4	s1	s9	s1	s9	s5	s9	s5	s5
c5	s1001	s1004	s1001	s1008	s1001	s1008	s1005	s1009	s1005	s1005
c6	s2006	s2006	s2006	s2006	s2006	s2006	s2006	s2006	s2006	s2006
Obj. Value	14.7	14.65	14.6	14.55	14.5	14.25	14.0	13.75	13.5	13.25
$E[UO_TSTT]$	14.95	14.9	14.850	15.050	14.750	14.80	14.450	14.3	14.15	14.0
$TSTT^{TARGET}$	14.883	14.767	14.65	14.533	14.417	14.3	14.183	14.067	13.95	13.833
TSTT1	20.70	20.650	20.600	21.050	20.500	20.650	20.000	19.750	19.500	19.250
$(Risk1)^{0.5}$	5.817	5.883	5.950	6.517	6.083	6.350	5.817	5.683	5.550	5.417
TSTT2	9.200	9.150	9.100	9.050	9.000	8.950	8.900	8.850	8.800	8.750
$(Risk2)^{0.5}$	0	0	0	0	0	0	0	0	0	0
$E[Risk]$	16.919	17.305	17.701	21.236	18.501	20.161	16.919	16.148	15.401	14.672

Table E.3. Solutions of BLPNDP-SSD at Different Total Budget Levels

BLP NDP-SSD	Total Available Budget									
	6	7	8	9	10	11	12	13	14	15
b1	1.6	2.2	2.8	3.4	4	0	0	5.333	6	6.667
b2	0	0	0	0	0	4.6	5	0	0	0
b3	4.4	4.8	5.2	5.6	6	6.4	7	7.667	8	8.333
b4	0	0	0	0	0	0	0	0	0	0
b5	0	0	0	0	0	0	0	0	0	0
b6	0	0	0	0	0	0	0	0	0	0
c3	s9	s8	s9	s8	s9	s11	s11	s9	s8	s8
c5	s1009	s1009	s1009	s1009	s1009	s1011	s1011	s1009	s1008	s1008
c6	s2006	s2006	s2006	s2010	s2006	s2006	s2010	s2010	s2010	s2010
Obj. Value	14.52	14.34	14.16	13.98	13.8	13.62	13.45	13.35	13.3	13.25
$E[UO_TSTT]$	14.93	15.06	14.690	14.820	14.450	14.330	14.450	14.3	14.2	14.083
$TSTT^{TARGET}$	14.883	14.767	14.65	14.533	14.417	14.3	14.183	14.067	13.95	13.833
TSTT1	20.520	20.840	20.160	20.480	19.800	19.620	19.950	19.750	19.600	19.417
(Risk1) ^{0.5}	5.637	6.073	5.510	5.947	5.383	5.320	5.767	5.683	5.650	5.584
TSTT2	9.340	9.280	9.220	9.160	9.100	9.040	8.950	8.850	8.800	8.750
(Risk2) ^{0.5}	0	0	0	0	0	0	0	0	0	0
$E[Risk]$	15.888	18.441	15.180	17.683	14.488	14.151	16.629	16.149	15.961	15.589

Table E.4. Solutions of SLP2-SONDP-SSD at Different Total Budget Levels

SLP2- SONDP-SSD	Total Available Budget									
	6	7	8	9	10	11	12	13	14	15
b1	0	0	0	0	0	0	0	5.333	0	0
b2	0.667	1.333	2	2.667	3.333	4	4.667	0	6	6.667
b3	5.333	5.667	6	6.333	6.667	7	7.333	7.667	8	8.333
b4	0	0	0	0	0	0	0	0	0	0
b5	0	0	0	0	0	0	0	0	0	0
b6	0	0	0	0	0	0	0	0	0	0
c3	s5	s11	s11	s11	s5	s11	s5	s9	s11	s11
c5	s1005	s1011	s1011	s1011	s1005	s1011	s1005	s1008	s1011	s1011
c6	s2006	s2006	s2006	s2006	s2006	s2006	s2006	s2006	s2006	s2006
Obj. Value	14.883	14.767	14.650	14.533	14.417	14.3	14.183	14.067	13.95	13.833
$E[UO_TSTT]$	14.883	14.767	14.65	14.533	14.417	14.30	14.183	14.067	13.95	13.833
$TSTT^{TARGET}$	14.883	14.767	14.65	14.533	14.417	14.3	14.183	14.067	13.95	13.833
TSTT1	20.567	20.383	20.200	20.017	19.833	19.650	19.467	19.283	19.100	18.917
(Risk1) ^{0.5}	5.684	5.616	5.550	5.484	5.416	5.350	5.284	5.216	5.150	5.084
TSTT2	9.200	9.150	9.100	9.050	9.000	8.950	8.900	8.850	8.800	8.750
(Risk2) ^{0.5}	0	0	0	0	0	0	0	0	0	0
$E[Risk]$	16.152	15.772	15.401	15.035	14.669	14.311	13.958	13.605	13.261	12.922

Table E.5. Solutions of SLP2-UONDP-SSD at Different Total Budget Levels

SLP2- UONDP-SSD	Total Available Budget									
	6	7	8	9	10	11	12	13	14	15
b1	0	0	0	0	0	1	2	3	4	0
b2	0	0	0	0	0	0	0	0	0	5
b3	6	7	8	9	10	10	10	10	10	10
b4	0	0	0	0	0	0	0	0	0	0
b5	0	0	0	0	0	0	0	0	0	0
b6	0	0	0	0	0	0	0	0	0	0
c3	s1	s1	s2	s1	s1	s9	s9	s8	s9	s5
c5	s1001	s1001	s1002	s1001	s1001	s1009	s1009	s1008	s1009	s1005
c6	s2006	s2006	s2006	s2006	s2006	s2006	s2006	s2006	s2006	s2006
Obj. Value	14.95	14.9	14.85	14.80	14.75	14.6	14.45	14.3	14.15	14
$E[UO_TSTT]$	14.95	14.9	14.85	14.80	14.75	14.6	14.45	14.3	14.15	14
$TSTT^{TARGET}$	14.883	14.767	14.65	14.533	14.417	14.3	14.183	14.067	13.95	13.833
TSTT1	20.700	20.650	20.600	20.550	20.500	20.250	20.000	19.750	19.500	19.250
(Risk1) ^{0.5}	5.817	5.883	5.950	6.017	6.083	5.950	5.817	5.683	5.550	5.417
TSTT2	9.200	9.150	9.100	9.050	9.000	8.950	8.900	8.850	8.800	8.750
(Risk2) ^{0.5}	0	0	0	0	0	0	0	0	0	0
$E[Risk]$	16.919	17.305	17.701	18.102	18.501	17.701	16.919	16.148	15.401	14.672

Table E.6. Solutions of SLP2-BLPNDP-SSD at Different Total Budget Levels

SLP2- BLPNDP-SSD	Total Available Budget									
	6	7	8	9	10	11	12	13	14	15
b1	0.667	0	0	2.667	0	0	0	0	0	6.667
b2	0	1.333	2	0	3.333	4	4.667	5.333	6	0
b3	5.333	5.667	6	6.333	6.667	7	7.333	7.667	8	8.333
b4	0	0	0	0	0	0	0	0	0	0
b5	0	0	0	0	0	0	0	0	0	0
b6	0	0	0	0	0	0	0	0	0	0
c3	s9	s11	s5	s8	s5	s5	s5	s11	s11	s9
c5	s1009	s1011	s1005	s1008	s1005	s1005	s1005	s1005	s1011	s1008
c6	s2006	s2006	s2006	s2006	s2006	s2006	s2006	s2006	s2006	s2006
Obj. Value	14.883	14.767	14.650	14.533	14.417	14.3	14.183	14.067	13.95	13.833
$E[UO_TSTT]$	14.883	14.767	14.65	14.533	14.417	14.3	14.183	14.067	13.95	13.833
$TSTT^{TARGET}$	14.883	14.767	14.65	14.533	14.417	14.3	14.183	14.067	13.95	13.833
TSTT1	20.567	20.383	20.200	20.017	19.833	19.650	19.467	19.283	19.100	18.917
(Risk1) ^{0.5}	5.684	5.616	5.550	5.484	5.416	5.350	5.284	5.216	5.150	5.084
TSTT2	9.200	9.150	9.100	9.050	9.000	8.950	8.900	8.850	8.800	8.750
(Risk2) ^{0.5}	0	0	0	0	0	0	0	0	0	0
$E[Risk]$	16.152	15.772	15.401	15.035	14.669	14.311	13.958	13.605	13.261	12.922

Table E.7. Solutions of RO-BLPNDP-SSD at Different Total Budget Levels

RO-BLPNDP-SSD ($\lambda=1-1000$)	Total Available Budget									
	6	7	8	9	10	11	12	13	14	15
b1	0	2.55	2.7	3.187	1.111	3.333	2.444	3.333	4	7
b2	1.6	0.2	0.8	1.062	3.333	1.778	3.333	3.111	3.333	0
b3	4.4	4.25	4.5	4.75	5.556	5.889	6.222	6.556	6.667	8
b4	0	0	0	0	0	0	0	0	0	0
b5	0	0	0	0	0	0	0	0	0	0
b6	0	0	0	0	0	0	0	0	0	0
c3	s11	s10	s10	s10	s12	s4	s7	s4	s11	s9
c5	s1011	s1006	s1010	s1006	s1012	s1002	s1001	s1004	s1005	s1009
c6	s2006	s2008	s2008	s2008	s2002	s2002	s2008	s2008	s2008	s2006
$E[UO_TSTT]$	14.930	15.009	14.847	14.703	14.556	14.439	14.322	14.206	14.083	13.850
$TSTT^{TARGET}$	14.883	14.767	14.65	14.533	14.417	14.3	14.183	14.067	13.95	13.833
TSTT1	20.520	20.292	20.045	19.831	19.667	19.517	19.367	19.217	19.067	18.900
(Risk1) ^{0.5}	5.637	5.525	5.395	5.298	5.250	5.217	5.184	5.150	5.117	5.067
TSTT2	9.340	9.725	9.650	9.575	9.444	9.361	9.278	9.194	9.100	8.800
(Risk2) ^{0.5}	0	0	0	0	0	0	0	0	0	0
$E[Risk]$	15.888	15.266	14.553	14.037	13.780	13.607	13.436	13.260	13.091	12.837

Appendix F. Factors of O-D Trip Matrix for Networks 4.1 and 4.2; and O-D Demand Realizations for Metaheuristics and Evaluation for Networks 3.2 and 4.2

Table F.1. Peak-Hour O-D Trip Factors for Network 4.1 (Table XI page 262 in Suwansirikul et al., 1987)

Origin	Destination											
	1	2	3	4	5	6	7	8	9	10	11	12
1	0	0.11	0.11	0.55	0.22	0.33	0.55	0.88	0.55	1.43	0.55	0.22
2	0.11	0	0.11	0.22	0.11	0.44	0.22	0.44	0.22	0.66	0.22	0.11
3	0.11	0.11	0	0.22	0.11	0.33	0.11	0.22	0.11	0.33	0.33	0.22
4	0.55	0.22	0.22	0	0.55	0.44	0.44	0.77	0.77	1.32	1.54	0.66
5	0.22	0.11	0.11	0.55	0	0.22	0.22	0.55	0.88	1.1	0.55	0.22
6	0.33	0.44	0.33	0.44	0.22	0	0.44	0.88	0.44	0.88	0.44	0.22
7	0.55	0.22	0.11	0.44	0.22	0.44	0	1.1	0.66	2.09	0.55	0.77
8	0.88	0.44	0.22	0.77	0.55	0.88	1.1	0	0.88	1.76	0.88	0.66
9	0.55	0.22	0.11	0.77	0.88	0.44	0.66	0.88	0	3.08	1.54	0.66
10	1.43	0.66	0.33	1.32	1.1	0.88	2.09	1.76	3.08	0	4.4	2.2
11	0.55	0.22	0.33	1.65	0.55	0.44	0.55	0.88	1.54	4.29	0	1.54
12	0.22	0.11	0.22	0.66	0.22	0.22	0.77	0.66	0.66	2.2	1.54	0
13	0.55	0.33	0.11	0.66	0.22	0.22	0.44	0.66	0.66	2.09	1.1	1.43
14	0.33	0.11	0.11	0.55	0.11	0.11	0.22	0.44	0.66	2.31	1.76	0.77
15	0.55	0.11	0.11	0.55	0.22	0.22	0.55	0.66	1.1	4.4	1.54	0.77
16	0.55	0.44	0.22	0.88	0.55	0.99	1.54	2.42	1.64	4.84	1.54	0.77
17	0.44	0.22	0.11	0.55	0.22	0.55	1.1	1.54	0.99	4.29	1.1	0.66
18	0.11	0	0	0.11	0	0.11	0.22	0.33	0.22	0.77	0.22	0.22
19	0.33	0.11	0	0.22	0.11	0.22	0.44	0.77	0.44	1.98	0.44	0.33
20	0.33	0.11	0	0.33	0.11	0.33	0.55	0.99	0.66	2.75	0.66	0.55
21	0.11	0	0	0.22	0.11	0.11	0.22	0.44	0.33	1.32	0.44	0.33
22	0.44	0.11	0.11	0.44	0.22	0.22	0.55	0.55	0.77	2.86	1.21	0.77
23	0.33	0	0.11	0.55	0.11	0.11	0.22	0.33	0.55	1.98	1.43	0.77
24	0.11	0	0	0.22	0	0.11	0.11	0.22	0.22	0.88	0.66	0.55

Origin	Destination											
	13	14	15	16	17	18	19	20	21	22	23	24
1	0.55	0.33	0.55	0.55	0.44	0.11	0.33	0.33	0.11	0.44	0.33	0.11
2	0.33	0.11	0.11	0.44	0.22	0	0.11	0.11	0	0.11	0	0
3	0.11	0.11	0.11	0.22	0.11	0	0	0	0	0.11	0.11	0
4	0.66	0.55	0.55	0.88	0.55	0.11	0.22	0.33	0.22	0.44	0.55	0.22
5	0.22	0.11	0.22	0.55	0.22	0	0.11	0.11	0.11	0.22	0.11	0
6	0.22	0.11	0.22	0.99	0.55	0.11	0.22	0.33	0.11	0.22	0.11	0.11
7	0.44	0.22	0.55	1.54	1.1	0.22	0.44	0.55	0.22	0.55	0.22	0.11
8	0.66	0.44	0.66	2.42	1.54	0.33	0.77	0.99	0.44	0.55	0.33	0.22
9	0.66	0.66	0.99	1.54	0.99	0.22	0.44	0.66	0.33	0.77	0.55	0.22
10	2.09	2.31	4.4	4.84	4.29	0.77	1.98	2.75	1.32	2.86	1.98	0.88
11	1.1	1.76	1.54	1.54	1.1	0.11	0.44	0.66	0.44	1.21	1.43	0.66
12	1.43	0.77	0.77	0.77	0.66	0.22	0.33	0.44	0.33	0.77	0.77	0.55
13	0	0.66	0.77	0.66	0.55	0.11	0.33	0.66	0.66	1.43	0.88	0.88
14	0.66	0	1.43	0.77	0.77	0.11	0.33	0.55	0.44	1.32	1.21	0.44
15	0.77	1.43	0	1.32	1.65	0.22	0.88	1.21	0.88	2.86	1.1	0.44
16	0.66	0.77	1.32	0	3.08	0.55	1.43	1.76	0.66	1.32	0.55	0.33
17	0.55	0.77	1.65	3.08	0	0.66	1.87	1.87	0.66	1.87	0.66	0.33
18	0.11	0.11	0.22	0.55	0.66	0	0.33	0.44	0.11	0.33	0.11	0
19	0.33	0.33	0.88	1.43	1.87	0.33	0	1.32	0.44	1.32	0.33	0.11
20	0.66	0.55	1.21	1.76	1.87	0.44	1.32	0	1.32	2.64	0.77	0.44
21	0.66	0.44	0.88	0.66	0.66	0.11	0.44	1.32	0	1.98	0.77	0.55
22	1.43	1.32	2.86	1.32	1.87	0.33	1.32	2.64	1.98	0	2.31	1.21
23	0.88	1.21	1.1	0.55	0.66	0.11	0.33	0.77	0.77	2.31	0	0.77
24	0.77	0.44	0.44	0.33	0.33	0	0.11	0.44	0.55	1.21	0.77	0

Note: The deterministic O-D matrix is determined by multiplying the factor matrix with 450 vph.

Table F.2. Peak-Hour O-D Trip Matrix for Network 4.2

Number	Origin	Destination	Factor	Deterministic Demand (Mean Value)
1	1	2	0.11	99
2	1	18	0.11	99
3	2	11	0.22	198
4	3	8	0.22	198
5	4	6	0.44	396
6	4	22	0.44	396
7	5	15	0.22	198
8	6	10	0.88	792
9	7	2	0.22	198
10	7	19	0.44	396
11	8	12	0.66	594
12	9	4	0.77	693
13	9	21	0.33	297
14	10	14	2.31	2079
15	11	6	0.44	396
16	11	23	1.43	1287
17	12	16	0.77	693
18	13	8	0.66	594
19	14	1	0.33	297
20	14	18	0.11	99
21	15	10	4.4	3960
22	16	3	0.22	198
23	16	20	1.76	1584
24	17	12	0.66	594
25	18	8	0.33	297
26	19	2	0.11	99
27	19	20	1.32	1188
28	20	13	0.66	594
29	21	8	0.44	396
30	22	1	0.44	396
31	22	17	1.87	1683
32	23	11	1.43	1287
33	24	7	0.11	99

Note: The demand is determined from multiplying the factor by a normal random variable (the mean of 900 vph and the standard deviation of 100).

Table F.3. 20 O-D Demand Realizations for Metaheuristics for Network 3.2 (12 O-D pairs)

O-D Demand Scenario	O-D											
	1-3	1-7	1-9	3-1	3-7	3-9	7-1	7-3	7-9	9-1	9-3	9-7
1	484	351	460	379	361	392	409	449	472	443	460	400
2	470	503	454	487	455	422	466	456	422	453	402	513
3	433	397	478	498	459	431	493	478	503	496	425	510
4	376	409	478	476	392	391	485	456	489	440	470	445
5	493	339	407	528	435	465	503	488	465	403	552	395
6	402	383	413	388	377	539	376	463	377	507	460	473
7	413	422	494	471	351	422	408	474	510	410	355	529
8	409	422	402	497	406	414	444	528	481	462	478	438
9	456	464	501	395	432	397	446	439	543	421	408	475
10	406	453	332	378	391	481	521	392	448	500	426	422
11	386	447	523	463	407	438	406	465	409	489	575	452
12	465	301	366	365	447	442	510	493	433	493	477	421
13	467	467	469	508	467	539	428	335	456	488	427	383
14	424	420	398	455	489	432	482	495	480	459	372	446
15	389	502	368	382	467	374	373	496	456	408	537	426
16	530	425	452	426	343	497	449	480	451	520	410	523
17	500	503	434	460	389	378	500	451	466	389	537	482
18	481	476	448	458	506	532	529	482	493	455	503	520
19	509	452	380	437	453	433	496	374	522	394	476	444
20	419	443	452	468	469	392	492	567	489	470	407	500

Table F.4. 30 O-D Demand Realizations for Evaluation for Network 3.2 (12 O-D pairs)

O-D Demand Scenario	O-D											
	1-3	1-7	1-9	3-1	3-7	3-9	7-1	7-3	7-9	9-1	9-3	9-7
1	439	448	413	397	570	394	385	486	433	367	439	406
2	475	354	531	456	449	430	406	594	419	495	417	375
3	461	417	379	483	469	432	418	504	472	544	405	432
4	394	446	491	365	433	409	508	430	497	424	412	483
5	400	423	440	452	413	513	457	416	505	435	380	396
6	465	467	562	485	460	544	507	473	490	548	444	468
7	463	428	525	517	510	439	458	470	502	551	482	506
8	468	451	524	471	435	504	496	364	453	374	441	428
9	533	453	424	514	401	396	443	440	390	436	444	488
10	454	412	433	475	403	469	483	351	436	487	472	437
11	437	476	434	391	499	433	383	503	509	533	399	434
12	544	473	462	456	400	452	368	383	417	546	462	448
13	436	418	408	480	399	369	429	401	439	443	513	442
14	523	397	460	454	429	464	483	371	443	367	492	547
15	509	460	446	495	463	455	403	463	400	453	490	377
16	421	407	379	481	503	490	483	366	454	441	459	477
17	420	413	442	565	448	424	454	458	506	495	403	416
18	498	477	410	398	470	521	466	437	490	536	330	441
19	459	436	512	423	454	480	403	376	452	450	495	474
20	505	462	497	463	410	443	342	423	451	431	378	443
21	412	461	474	476	414	454	482	453	516	537	456	438
22	372	416	448	489	392	444	397	520	460	369	454	410
23	478	440	507	389	523	406	448	531	332	422	382	356
24	566	409	485	484	462	507	424	443	443	433	489	482
25	479	424	469	420	495	449	454	428	399	478	623	472
26	385	473	442	400	446	388	481	529	398	493	433	438
27	511	405	459	450	444	535	479	366	438	504	398	447
28	371	430	488	440	446	495	377	522	481	505	452	479
29	390	429	541	392	523	520	464	410	386	405	452	461
30	488	423	454	451	500	420	427	484	446	418	404	341

Table F.5. 20 O-D Demand Realizations for Metaheuristics for Network 4.2 (33 O-D pairs)

O-D	O-D Demand Scenario																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1-2	95	108	83	115	75	89	106	92	103	105	79	104	89	113	93	106	92	104	106	83
1-18	118	98	100	107	82	85	97	108	94	107	102	104	78	118	94	104	117	105	99	100
2-11	261	187	199	192	174	221	177	204	171	195	175	232	208	175	168	242	192	212	200	176
3-8	178	203	186	190	167	226	218	178	178	258	216	211	218	199	211	231	205	188	235	195
4-6	393	453	337	346	350	407	424	416	410	422	446	431	335	442	354	437	379	387	439	325
4-22	471	410	318	432	414	441	440	344	446	441	368	347	444	455	373	329	484	402	296	399
5-15	183	175	161	211	168	201	184	183	195	184	175	212	210	196	173	175	223	244	205	198
6-10	596	856	761	812	638	833	869	845	943	859	757	834	720	896	1008	684	792	805	698	991
7-2	210	213	195	149	209	227	199	217	182	175	200	182	210	236	179	195	229	202	218	237
7-19	530	373	355	412	376	405	408	341	378	406	403	414	438	360	365	316	435	383	427	366
8-12	575	556	598	483	579	671	755	634	556	599	576	510	521	550	539	613	585	647	526	669
9-4	876	756	714	549	574	790	719	794	723	688	510	744	636	661	674	722	654	743	574	723
9-21	273	293	290	320	318	329	297	280	310	330	362	291	312	317	321	293	331	271	281	267
10-14	1987	1662	1989	1994	2136	2292	2559	1892	2178	1253	1937	2366	2295	1972	1849	1934	2316	2066	2123	2268
11-6	474	383	434	399	487	437	454	398	432	351	439	340	435	471	373	418	366	409	348	370
11-23	1271	1488	1158	1233	1307	1101	1144	1112	1082	1394	874	1152	1342	1391	1287	976	1289	1355	1413	1352
12-16	630	771	638	722	650	650	669	654	694	746	699	721	755	609	724	758	559	742	759	799
13-8	597	547	710	542	650	623	564	534	573	599	625	490	612	607	518	529	567	602	565	601
14-1	308	309	308	299	327	340	280	273	280	327	305	348	352	310	283	323	254	239	317	279
14-18	114	91	98	114	82	109	89	104	76	104	117	86	107	92	114	108	96	107	98	102
15-10	3510	3829	3972	3629	3996	3829	3744	3970	3971	4653	3962	3648	3760	4049	3740	4129	3573	4567	3700	3620
16-3	220	250	179	169	190	215	204	215	225	211	199	197	173	222	195	178	193	199	171	205
16-20	1600	1643	1538	1734	1592	1358	1382	1529	1782	1672	1887	1502	1483	1613	1255	1417	1728	1491	1655	1381
17-12	574	579	600	563	594	682	593	620	654	552	495	505	561	703	739	703	550	482	517	571
18-8	376	309	315	342	285	286	284	390	284	291	323	235	256	292	265	283	249	246	346	266
19-2	96	87	99	91	77	105	109	110	123	126	108	91	109	114	101	89	101	88	98	106
19-20	1361	1002	1193	926	1430	1335	1244	1443	1211	1250	1285	1203	1437	1287	1158	1118	1223	1233	1200	1312
20-13	487	710	643	568	496	579	571	669	550	687	476	736	603	604	618	556	620	726	538	609
21-8	436	418	371	418	367	412	420	359	388	392	351	445	393	328	386	389	428	374	438	461
22-1	385	386	418	286	398	427	415	401	383	391	372	384	443	389	401	391	380	400	441	352
22-17	1557	1853	1490	1571	1697	1788	1415	2141	1581	2076	1464	1766	1559	1508	1556	1632	1808	1898	1809	1574
23-11	1507	1351	1299	994	1077	1024	1220	1155	1358	1243	1200	1208	1277	1353	1398	1197	1151	1203	1378	1193
24-7	108	115	93	112	82	89	117	92	105	115	75	95	91	98	110	99	107	86	100	110

Table F.6. 30 O-D Demand Realizations for Evaluation for Network 4.2 (33 O-D pairs)

O-D	O-D Demand Scenario														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1-2	95	90	104	97	109	109	103	98	88	111	95	98	95	102	93
1-18	91	103	114	106	97	108	110	78	71	106	63	98	67	104	90
2-11	156	189	186	211	194	204	211	200	192	193	195	214	200	171	234
3-8	201	162	211	172	240	195	219	195	179	159	193	194	246	195	198
4-6	335	361	405	381	362	397	363	398	362	366	442	481	434	376	305
4-22	363	357	355	457	395	472	464	434	333	336	373	439	402	380	312
5-15	216	196	187	169	211	186	162	218	171	208	265	191	180	199	232
6-10	877	797	793	790	759	811	777	629	696	963	672	779	789	727	789
7-2	192	220	180	146	218	216	187	208	152	207	223	189	209	196	201
7-19	350	390	354	355	383	376	389	393	440	384	365	341	384	379	428
8-12	581	691	656	485	683	631	447	690	569	589	627	591	657	555	498
9-4	741	713	549	729	633	694	602	627	644	886	620	744	672	724	652
9-21	264	307	272	240	274	280	338	227	362	327	372	333	248	332	257
10-14	2432	2048	2036	1806	2201	1744	2533	2512	2118	1968	2213	2395	2261	2171	1974
11-6	353	428	326	416	443	456	378	410	458	341	385	383	392	439	379
11-23	1516	1493	1151	1245	1359	1227	1325	1189	1315	1350	1473	1179	1353	1135	1306
12-16	671	837	810	716	697	619	684	803	705	774	738	750	574	853	676
13-8	678	599	673	533	764	470	662	626	595	629	575	666	555	543	563
14-1	344	277	299	352	338	300	343	297	290	271	277	279	256	275	317
14-18	98	104	105	113	97	115	98	110	105	107	81	108	96	107	95
15-10	3574	4358	4492	4026	4119	4588	4090	3506	3589	3800	3641	4159	4745	3828	4781
16-3	186	198	195	217	205	202	203	217	180	218	184	213	276	223	202
16-20	1442	1429	1723	1765	1499	1273	1559	1783	1707	1465	1577	1261	1494	1774	1437
17-12	668	615	660	585	552	498	553	664	701	581	699	652	546	639	528
18-8	287	340	303	260	284	320	260	334	278	298	307	275	278	271	342
19-2	90	78	104	106	90	91	97	101	90	88	103	95	101	130	99
19-20	1327	1049	1057	1315	1150	1127	1205	1323	1285	1343	1108	899	1273	1097	1280
20-13	631	618	602	555	528	664	638	555	590	657	555	575	556	635	492
21-8	363	436	465	456	462	434	375	435	407	433	418	504	391	406	381
22-1	444	392	488	463	301	403	395	304	365	312	396	411	372	449	385
22-17	1794	1605	1761	1740	1848	1428	1333	1961	1639	1880	1776	1701	1382	1461	1382
23-11	1304	1148	1283	1398	1366	1141	1258	1318	1356	1261	1217	1226	1248	1380	1596
24-7	94	111	96	108	102	97	116	108	99	100	117	112	100	88	78

Table F.6. 30 O-D Demand Realizations for Evaluation for Network 4.2 (33 O-D pairs) (continued)

O-D	O-D Demand Scenario														
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1-2	110	122	95	89	97	95	106	105	101	90	89	85	96	86	101
1-18	112	109	93	96	102	79	83	94	119	119	94	106	85	101	83
2-11	212	206	186	201	220	176	182	220	185	177	249	199	212	163	213
3-8	191	204	220	142	193	183	209	215	195	180	199	205	163	176	201
4-6	473	423	398	423	418	378	397	404	337	436	364	342	353	408	389
4-22	401	362	391	479	358	509	402	445	373	394	378	363	403	367	406
5-15	217	189	184	186	199	193	161	183	191	223	229	203	237	200	171
6-10	721	919	889	667	751	796	708	800	965	750	893	859	823	745	806
7-2	226	168	168	185	183	209	170	154	166	215	191	152	202	189	150
7-19	390	405	428	384	337	348	466	317	454	388	382	384	384	367	460
8-12	553	558	588	500	547	558	634	494	645	565	634	560	613	506	606
9-4	824	763	588	656	720	508	704	782	791	773	765	619	613	726	588
9-21	301	257	295	327	315	311	295	375	345	305	307	299	292	310	246
10-14	2063	1919	1894	2089	2022	1990	2069	1754	1766	1858	1835	2168	1902	1740	2254
11-6	363	385	373	418	388	453	311	389	463	456	418	433	397	464	370
11-23	1267	1337	1389	1454	1024	1575	1201	1315	1178	1518	1226	1153	1164	1249	988
12-16	627	758	727	796	633	757	833	595	650	738	744	667	654	698	707
13-8	664	753	725	550	645	518	602	552	481	527	649	659	495	475	483
14-1	264	305	382	326	276	342	225	256	314	325	311	288	257	345	352
14-18	102	90	97	96	95	103	118	110	102	102	86	113	107	83	104
15-10	4071	3334	3685	3742	3988	3830	3679	4144	4242	3870	3670	3858	3490	4295	4115
16-3	230	222	182	206	224	228	220	220	174	185	199	230	203	213	205
16-20	1495	1397	1554	1939	1659	1521	1545	1530	1372	1311	1580	1388	1765	1363	1808
17-12	330	557	549	515	695	623	664	588	634	640	480	647	635	710	571
18-8	316	258	313	272	309	304	288	275	343	348	365	293	303	359	318
19-2	81	100	93	105	84	95	111	101	96	90	115	104	82	94	108
19-20	1069	1148	1434	973	971	1112	1447	1191	1224	1119	1179	1460	1233	1158	934
20-13	630	577	574	689	655	568	678	565	488	715	521	637	605	520	571
21-8	358	300	399	363	410	465	471	418	407	418	386	367	331	348	337
22-1	410	312	413	332	326	407	419	406	323	422	403	494	341	429	429
22-17	1407	1956	1614	1524	1902	1511	1728	1974	1499	1773	1587	1813	1667	1798	1625
23-11	1479	1210	1209	1201	1410	1589	1147	1469	1221	1268	1264	1513	970	934	1142
24-7	83	93	100	101	122	110	101	103	120	97	102	89	113	105	106

Appendix G. Detailed Solutions from Metaheuristics in Section 6.5

Table G.1. Solutions of DET-SSD, SP-SSD and RO-SSD on Network 3.2 (12 O-D pairs); 100 Trials, 20 samples of O-D demands, $TSTT^{TARGET}=575.7428$

	DET-SSD	SP-SSD	RO-SSD
Cycle:			
Node 5	196	96	130
Node 4	123	177	56
Node 2	163	77	97
Offset:			
Node 5	42	75	56
Node 4	11	93	21
Node 2	46	45	32
Phase Seq:			
Node 5	5	10	3
Node 4	0	0	1
Node 2	1	0	0
Phase Green			
Node 5	22, 57, 54, 39	10, 25, 19, 18	26, 32, 27, 21
Node 4	25, 29, 51	21, 111, 27	17, 10, 11
Node 2	110, 11, 24	17, 30, 12	15, 47, 17
Trial found	64	54	99
CPU (hrs)	1.585	41.099	40.5528

Table G.2. Solutions of DET-NDP, SP-NDP and RO-NDP on Network 3.2 (12 O-D pairs); 100 Trials, 20 samples of O-D demands, $TSTT^{TARGET}=575.7428$

Link	DET-NDP	SP-NDP	RO-NDP
12	55.8376	99.2018	101.6636
21	76.1421	18.8141	72.0887
23	55.8376	106.8985	12.939
32	67.2589	27.366	22.1811
45	82.4873	50.4561	2.7726
54	63.4518	17.1038	111.8299
56	64.7208	82.9532	91.4972
65	58.3756	94.0707	116.451
78	25.3807	88.0844	71.1645
87	55.8376	94.9259	86.8762
89	12.6904	85.5188	110.9057
98	1.269	105.1881	101.6636
14	26.6497	84.6636	84.1035
41	128.1726	57.2976	4.6211
25	17.7665	106.8985	12.0148
52	78.6802	96.6363	7.3937
36	67.2589	1.7104	4.6211
63	157.3604	25.6556	29.5749
47	1.269	12.8278	84.1035
74	74.8731	33.3523	117.3752
58	64.7208	83.8084	16.6359
85	159.8985	35.0627	66.5434
69	96.4467	39.3387	98.8909
96	7.6142	52.1665	72.0887
Trial found	60	75	100
CPU (hrs)	0.776	28.9984	28.9466

Table G.3. Solutions of DET-NDP-SSD, SP-NDP-SSD and RO-NDP-SSD on Network 3.2 (12 O-D pairs); 200 Trials, 20 samples of O-D demands, $TSTT^{TARGET} = 575.7428$

	DET-NDP-SSD	SP-NDP-SSD	RO-NDP-SSD
Cycle:			
Node 5	75	79	105
Node 4	92	76	69
Node 2	62	89	60
Offset:			
Node 5	7	70	4
Node 4	76	62	23
Node 2	9	83	57
Phase Seq:			
Node 5	13	11	7
Node 4	0	1	0
Node 2	0	1	0
Phase Green			
Node 5	10, 15, 10, 16	11, 16, 11, 17	34, 19, 11, 17
Node 4	26, 17, 31	23, 16, 19	17, 22, 12
Node 2	15, 11, 18	31, 19, 21	15, 11, 16

Link	DET-NDP-SSD	SP-NDP	RO-NDP
12	116.2362	19.1304	63.786
21	23.9852	33.913	52.4691
23	84.8708	107.8261	31.893
32	19.3727	86.087	23.6626
45	106.0886	40.8696	34.9794
54	44.2804	43.4783	120.3704
56	106.0886	21.7391	104.9383
65	57.1956	85.2174	58.642
78	3.69	92.1739	49.3827
87	99.631	52.1739	62.7572
89	81.1808	100	59.6708
98	96.8635	66.9565	19.5473
14	50.738	79.1304	13.3745
41	89.4834	49.5652	101.8519
25	13.8376	90.4348	4.1152
52	53.5055	26.9565	60.6996
36	50.738	71.3043	29.8354
63	41.5129	3.4783	22.6337
47	24.9077	80	93.6214
74	104.2435	60	60.6996
58	60.8856	29.5652	103.9095
85	101.476	100	77.1605
69	13.8376	93.913	127.572
96	55.3506	66.087	122.428
Trial found	165	182	198
CPU (hrs)	3.7670	86.1975	84.6905

Table G.4. Solutions of DET-NDP-D, SP-NDP-D and RO-NDP-D on Network 3.2 (12 O-D pairs); 75 Trials, 20 samples of O-D demands, $TSTT^{TARGET}=575.7428$

Link	DET-NDP-D	SP-NDP-D	RO-NDP-D
12	97.4923	114.6845	113.963
21	94.4217	28.216	63.655
23	93.654	24.5752	30.8008
32	13.0502	19.1141	4.1068
45	50.6653	60.983	100.616
54	95.1894	30.9466	30.8008
56	94.4217	104.6723	118.0698
65	35.3122	101.9417	78.0287
78	22.262	6.3714	14.3737
87	75.2303	92.8398	128.3368
89	18.4237	80.0971	84.1889
98	97.4923	96.4806	123.2033
14	3.0706	103.7621	33.8809
41	77.5333	73.7257	37.9877
25	71.392	8.1917	108.8296
52	92.8864	114.6845	92.4025
36	40.6858	25.4854	22.5873
63	84.4422	18.2039	10.2669
47	62.1801	102.8519	9.2402
74	90.5834	30.9466	52.3614
58	52.2006	101.9417	32.8542
85	73.695	85.5583	96.5092
69	36.0798	39.1383	80.0821
96	27.6356	34.5874	32.8542
Trial found	61	67	74
CPU (hrs)	1.1984	37.9824	26.3586

Table G.5. Solutions of DET-NDP-D, SP-NDP-D and RO-NDP-D on Network 4.2 (33 O-D pairs); 75 Trials, 20 samples of O-D demands, $TSTT^{TARGET} = 3405.5906$

Link	DET-NDP-D	SP-NDP-D	RO-NDP-D
910	271.6895	143.531	184.9315
109	95.8904	236.5229	5.137
68	287.6712	196.0916	282.5342
86	47.9452	256.7385	241.4384
87	223.7443	222.372	7.7055
78	54.7945	22.2372	102.7397
1016	198.6301	12.1294	308.2192
1610	100.4566	117.2507	69.3493
1324	9.1324	86.9272	200.3425
2413	210.0457	206.1995	97.6027
Trial found	14	22	71
CPU (hrs)	7.3039	92.9395	92.0412

Table G.6. Solutions of DET-SSD, SP-SSD and RO-SSD on Network 4.2 (33 O-D pairs); 75 Trials, 20 samples of O-D demands, $TSTT^{TARGET} = 3405.5906$

	DET-SSD	SP-SSD	RO-SSD
Cycle:			
Node 10	204	86	160
Node 8	124	109	204
Node 16	177	146	196
Node 6	169	140	97
Node 9	134	56	49
Node 17	53	72	56
Node 18	93	64	107
Offset:			
Node 10	181	62	144
Node 8	20	15	4
Node 16	68	101	189
Node 6	79	84	6
Node 9	88	47	23
Node 17	16	44	27
Node 18	30	61	7
Phase Seq:			
Node 10	9	11	5
Node 8	13	9	15
Node 16	9	15	4
Node 6	0	0	0
Node 9	1	1	1
Node 17	1	1	0
Node 18	1	1	0
Phase Green			
Node 10	19, 13, 16, 45, 81	10, 10, 11, 11, 14	11, 16, 46, 20, 37
Node 8	24, 13, 45, 18	23, 13, 10, 39	20, 30, 88, 42
Node 16	101, 14, 13, 25	20, 16, 55, 31	31, 35, 14, 92
Node 6	71, 30, 50	13, 19, 90	15, 10, 54
Node 9	31, 38, 47	11, 11, 16	10, 10, 11
Node 17	11, 12, 12	12, 22, 20	13, 14, 11
Node 18	13, 15, 47	12, 22, 12	10, 14, 65
Trial found	69	10	71
CPU (hrs)	7.3218	30.998	90.9297

Table G.7. Solutions of DET-NDP, SP-NDP and RO-NDP on Network 4.2 (33 O-D pairs); 75 Trials, 20 samples of O-D demands, $TSTT^{TARGET} = 3405.5906$

Link	DET-NDP	SP-NDP	RO-NDP
910	12.2616	205.6673	203.7643
109	98.0926	172.7605	2.455
68	188.0109	21.9378	159.5745
86	161.4441	301.6453	149.7545
87	251.3624	5.4845	144.8445
78	114.4414	205.6673	36.8249
1016	235.0136	326.3254	149.7545
1610	87.8747	230.3473	274.9591
1324	134.8774	13.7112	238.1342
2413	216.6213	16.4534	139.9345
Trial found	72	10	72
CPU (hrs)	2.8581	27.2525	31.5779

Table G.8. Solutions of DET-NDP-SSD, SP-NDP-SSD and RO-NDP-SSD on Network 4.2 (33 O-D pairs); 150 Trials, 20 samples of O-D demands, $TSTT^{TARGET} = 3405.5906$

	DET-NDP-SSD	SP-NDP-SSD	RO-NDP-SSD
Cycle:			
Node 10	200	229	240
Node 8	103	132	135
Node 16	109	123	126
Node 6	76	61	128
Node 9	64	107	73
Node 17	95	72	77
Node 18	138	61	57
Offset:			
Node 10	91	99	119
Node 8	90	31	29
Node 16	16	119	106
Node 6	49	58	123
Node 9	36	7	41
Node 17	62	31	32
Node 18	47	4	11
Phase Seq:			
Node 10	2	3	3
Node 8	0	5	5
Node 16	13	0	8
Node 6	0	1	0
Node 9	1	0	0
Node 17	0	0	1
Node 18	1	0	1
Phase Green			
Node 10	64, 10, 12, 45, 39	29, 10, 21, 57, 82	27, 10, 11, 54, 108
Node 8	11, 13, 10, 45	21, 13, 10, 64	25, 12, 11, 63
Node 16	21, 25, 21, 18	42, 23, 15, 19	38, 28, 24, 12
Node 6	11, 24, 23	10, 12, 21	11, 12, 87
Node 9	12, 12, 22	13, 26, 50	12, 14, 29
Node 17	29, 30, 18	11, 26, 17	16, 24, 19
Node 18	12, 30, 78	16, 10, 17	14, 10, 15

Link	DET-NDP-SSD	SP-NDP-SSD	RO-NDP-SSD
910	86.3095	259.6154	224.8157
109	104.1667	193.6813	143.7346
68	44.6429	61.8132	47.9115
86	133.9286	113.3242	99.5086
87	11.9048	230.7692	210.0737
78	232.1429	98.9011	206.3882
1016	235.119	187.5	226.6585
1610	139.881	243.1319	210.0737
1324	232.1429	28.8462	27.6413
2413	279.7619	82.4176	103.1941
Trial found	142	97	125
CPU (hrs)	18.4408	131.6923	131.8564

Appendix H. Robustness Evaluation Results of Deterministic, Stochastic and Robust Solutions from Metaheuristics in Section 6.5

The new set of 30 O-D demand realizations in Tables F.4 and F.6 for Networks 3.2 and 4.2 is used for all evaluations to prevent a possible bias. It is recalled that the 20 O-D demand realizations in Tables F.3 and F.5 for Networks 3.2 and 4.2 are employed in the metaheuristic searches. To calculate two-sided 95% CIs, we use the student-t statistics $t(0.025, 29)=2.045$.

Table H.1. Robustness Evaluation Results of Original Networks 3.2 and 4.2

	Original Network 3.2	Original Network 4.2
$E[Robust_Obj(\lambda=1)]$	3397.8	230841
Half-width of 95% CI of $Robust_Obj(\lambda=1)$	1301.5924	63543.1
95%CI = [lower, upper]: lower	2096.2076	167297.9
Upper	4699.3924	294384.1
SD of $Robust_Obj(\lambda=1)$	3486.12	170191
$E[TSTT]$	620.705	4054.96
Half-width of 95% CI of $TSTT$	10.0969	81.6152
95%CI = [lower, upper]: lower	610.6081	3973.3448
Upper	630.8019	4136.5752
SD of $TSTT$	27.0429	218.594
$E[Risk]$	2777.09	226787
Half-width of 95% CI of $Risk$	1292.1015	63468.182
95%CI = [lower, upper]: lower	1484.9885	163318.818
Upper	4069.1915	290255.182
SD of $Risk$	3460.7	169990
$V[TSTT]$	731.32	47783.3
Expected number of total vehicle trips	5406.6	22341.5
Expected total free flow travel time ($E[TFFT]$)	580.427	1514.037
$E[TSTT]/E[TFFT]$	1.0694	2.6782

Note: $E[TFFT]$ is determined by first calculating $TFFT$ for each O-D demand realization.

$TFFT$ is computed by assuming that all vehicles travel at free flow speed. Let the weight of each demand realization be to the number of vehicle trips; then, $E[TFFT]$ is the weighted average of $TFFT$.

Table H.2. Robustness Evaluation Results of Deterministic, Stochastic and Robust NDP-D, SSD, NDP, NDP-SSD Solutions on Network 3.2

a) Robustness Evaluation of NDP-D Solutions

	DET-NDP-D	SP-NDP-D	RO-NDP-D
$E[Robust_Obj(\lambda=1)]$	2680.27	2193.33	2150.25
Half-width of 95% CI of $Robust_Obj(\lambda=1)$	1183.85	1012.29	902.324
95%CI = [lower, upper]: lower	1496.42	1181.04	1247.926
Upper	3864.12	3205.62	3052.574
SD of $Robust_Obj(\lambda=1)$	3170.76	2711.27	2416.74
$E[TSTT]$	612.004	605.513	606.116
Half-width of 95% CI of $TSTT$	10.1549	9.83881	9.21964
95%CI = [lower, upper]: lower	601.8491	595.67419	596.89636
Upper	622.1589	615.35181	615.33564
SD of $TSTT$	27.1984	26.3518	24.6934
$E[Risk]$	2068.27	1587.82	1544.13
Half-width of 95% CI of $Risk$	1174.2640	1003.2446	893.7592
95%CI = [lower, upper]: lower	894.0060	584.5754	650.3708
Upper	3242.5340	2591.0646	2437.8892
SD of $Risk$	3145.09	2687.04	2393.8
$V[TSTT]$	739.752	694.416	609.765
$E[TSTT]/E[TFFT]$	1.0544	1.0432	1.0443

b) Robustness Evaluation of SSD Solutions

	DET-SSD	SP-SSD	RO-SSD
$E[Robust_Obj(\lambda=1)]$	3620.92	1514.1	1343.76
Half-width of 95% CI of $Robust_Obj(\lambda=1)$	1240.24	637.276	516.129
95%CI = [lower, upper]: lower	2380.68	876.824	827.631
Upper	4861.16	2151.376	1859.889
SD of $Robust_Obj(\lambda=1)$	3321.8	1706.85	1382.37
$E[TSTT]$	622.997	594.87	591.942
Half-width of 95% CI of $TSTT$	10.1471	8.94929	8.47403
95%CI = [lower, upper]: lower	612.8499	585.92071	583.46797
Upper	633.1441	603.81929	600.41603
SD of $TSTT$	27.1774	23.9693	22.6964
$E[Risk]$	2997.92	919.232	751.822
Half-width of 95% CI of $Risk$	1230.6159	629.3240	508.7087
95%CI = [lower, upper]: lower	1767.3041	289.9080	243.1133
Upper	4228.5359	1548.5560	1260.5307
SD of $Risk$	3296.02	1685.55	1362.5
$V[TSTT]$	738.61	574.529	515.128
$E[TSTT]/E[TFFT]$	1.0733	1.0249	1.0198

c) Robustness Evaluation of NDP Solutions

	DET-NDP	SP-NDP	RO-NDP
$E[Robust_Obj(\lambda=1)]$	2438.13	1334.22	1094.3
Half-width of 95% CI of $Robust_Obj(\lambda=1)$	964.485	554.857	417.047
95%CI = [lower, upper]: lower	1473.645	779.363	677.253
Upper	3402.615	1889.077	1511.347
SD of $Robust_Obj(\lambda=1)$	2583.23	1486.1	1117
$E[TSTT]$	610.269	591.428	585.421
Half-width of 95% CI of $TSTT$	9.30613	8.50332	8.15918
95%CI = [lower, upper]: lower	600.96287	582.92468	577.26182
Upper	619.57513	599.93132	593.58018
SD of $TSTT$	24.9251	22.7749	21.8531
$E[Risk]$	1827.86	742.79	508.876
Half-width of 95% CI of $Risk$	955.8348	547.3295	410.0472
95%CI = [lower, upper]: lower	872.0252	195.4605	98.8288
Upper	2783.6948	1290.1195	918.9232
SD of $Risk$	2560.06	1465.94	1098.25
$V[TSTT]$	621.259	518.694	477.56
$E[TSTT]/E[TFFT]$	1.0514	1.0190	1.0086

d) Robustness Evaluation of NDP-SSD Solutions

	DET-NDP-SSD	SP-NDP-SSD	RO-NDP-SSD
$E[Robust_Obj(\lambda=1)]$	2000.19	1243.55	992.93
Half-width of 95% CI of $Robust_Obj(\lambda=1)$	1029.01	558.03	378.13
95%CI = [lower, upper]: lower	971.18	685.52	614.8
Upper	3029.2	1801.58	1371.06
SD of $Robust_Obj(\lambda=1)$	2756.04	1494.61	1012.78
$E[TSTT]$	600.37	589.44	581.87
Half-width of 95% CI of $TSTT$	10.59	8.33	8.12
95%CI = [lower, upper]: lower	589.78	581.11	573.75
Upper	610.96	597.77	589.99
SD of $TSTT$	28.37	22.31	21.75
$E[Risk]$	1399.83	654.11	411.06
Half-width of 95% CI of $Risk$	1019.4038	550.8354	371.4899
95%CI = [lower, upper]: lower	380.4262	103.2746	39.5701
Upper	2419.2338	1204.9454	782.5499
SD of $Risk$	2730.32	1475.33	994.98
$V[TSTT]$	805.07	497.73	473
$E[TSTT]/E[TFFT]$	1.0344	1.0155	1.0025

Table H.3. Robustness Evaluation Results of Deterministic, Stochastic and Robust NDP-D, SSD, NDP, NDP-SSD Solutions on Network 4.2

a) Robustness Evaluation of NDP-D Solutions

	DET-NDP-D	SP-NDP-D	RO-NDP-D
$E[Robust_Obj(\lambda=1)]$	231892	220361	217679
Half-width of 95% CI of $Robust_Obj(\lambda=1)$	75594.1	66306.4	63889.1
95%CI = [lower, upper]: lower	156297.9	154054.6	153789.9
Upper	307486.1	286667.4	281568.1
SD of $Robust_Obj(\lambda=1)$	202468	177592	171117
$E[TSTT]$	4041.78	4036.81	4037.5
Half-width of 95% CI of $TSTT$	92.0368	85.8439	82.9764
95%CI = [lower, upper]: lower	3949.7432	3950.9661	3954.5236
Upper	4133.8168	4122.6539	4120.4764
SD of $TSTT$	246.507	229.92	222.24
$E[Risk]$	227850	216324	213641
Half-width of 95% CI of $Risk$	75509.1778	66226.5968	63812.0505
95%CI = [lower, upper]: lower	152340.8222	150097.4032	149828.9495
Upper	303359.1778	282550.5968	277453.0505
SD of $Risk$	202240	177378	170911
$V[TSTT]$	60765.6	52863.3	49390.6
$E[TSTT]/E[TFFT]$	2.6695	2.6663	2.6667

b) Robustness Evaluation of SSD Solutions

	DET-SSD	SP-SSD	RO-SSD
$E[Robust_Obj(\lambda=1)]$	108786	72614.9	56343.6
Half-width of 95% CI of $Robust_Obj(\lambda=1)$	63072.1	35196.5	26890.9
95%CI = [lower, upper]: lower	45713.9	37418.4	29452.7
Upper	171858.1	107811.4	83234.5
SD of $Robust_Obj(\lambda=1)$	168929	94268.4	72023.1
$E[TSTT]$	3822.56	3799.07	3754.08
Half-width of 95% CI of $TSTT$	102.952	82.352	84.7733
95%CI = [lower, upper]: lower	3719.608	3716.718	3669.3067
Upper	3925.512	3881.422	3838.8533
SD of $TSTT$	275.742	220.568	227.052
$E[Risk]$	104964	68815.8	52589.5
Half-width of 95% CI of $Risk$	62984.6754	35127.5983	26822.0741
95%CI = [lower, upper]: lower	41979.3246	33688.2017	25767.4259
Upper	167948.6754	103943.3983	79411.5741
SD of $Risk$	168695	94084	71838.9
$V[TSTT]$	76033.5	48650	51552.8
$E[TSTT]/E[TFFT]$	2.5247	2.5092	2.4795

c) Robustness Evaluation of NDP Solutions

	DET-NDP	SP-NDP	RO-NDP
$E[Robust_Obj(\lambda=1)]$	76870.6	72506.3	56196.3
Half-width of 95% CI of $Robust_Obj(\lambda=1)$	30802.2	34992.3	27473.6
95%CI = [lower, upper]: lower	46068.4	37514	28722.7
Upper	107672.8	107498.6	83669.9
SD of $Robust_Obj(\lambda=1)$	82499.1	93721.5	73583.9
$E[TSTT]$	3816.77	3794.85	3760.19
Half-width of 95% CI of $TSTT$	76.8409	85.8217	78.3094
95%CI = [lower, upper]: lower	3739.9291	3709.0283	3681.8806
Upper	3893.6109	3880.6717	3838.4994
SD of $TSTT$	205.807	229.861	209.74
$E[Risk]$	73053.9	68711.5	52436.1
Half-width of 95% CI of $Risk$	30732.8776	34922.3974	27407.5466
95%CI = [lower, upper]: lower	42321.0224	33789.1026	25028.5534
Upper	103786.7776	103633.8974	79843.6466
SD of $Risk$	82313.4	93534.4	73407
$V[TSTT]$	42356.5	52835.9	43990.9
$E[TSTT]/E[TFFT]$	2.5209	2.5064	2.4836

d) Robustness Evaluation of NDP-SSD Solutions

	DET-NDP-SSD	SP-NDP-SSD	RO-NDP-SSD
$E[Robust_Obj(\lambda=1)]$	80175.5	61694.8	46267.5
Half-width of 95% CI of $Robust_Obj(\lambda=1)$	45282.1	26092.3	24416.4
95%CI = [lower, upper]: lower	34893.4	35602.5	21851.1
Upper	125457.6	87787.1	70683.9
SD of $Robust_Obj(\lambda=1)$	121281	69884.3	65395.6
$E[TSTT]$	3807.75	3770.48	3737.05
Half-width of 95% CI of $TSTT$	84.265	86.3266	73.2525
95%CI = [lower, upper]: lower	3723.485	3684.1534	3663.7975
Upper	3892.015	3856.8066	3810.3025
SD of $TSTT$	225.691	231.213	196.196
$E[Risk]$	76367.8	57924.3	42530.4
Half-width of 95% CI of $Risk$	45211.7923	26023.7468	24355.6675
95%CI = [lower, upper]: lower	31156.0077	31900.5532	18174.7325
Upper	121579.5923	83948.0468	66886.0675
SD of $Risk$	121093	69700.7	65233
$V[TSTT]$	50936.5	53459.4	38492.8
$E[TSTT]/E[TFFT]$	2.5150	2.4903	2.4683

Appendix I. Paired *t* Tests in Section 6.5

Table I.1. *TSTT* of 30 OD Demand Scenarios for Network 3.2

Scenario	Problem (see note)												
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	626.07	584.59	569.72	579.27	633.06	585.94	566.86	599.56	569.11	560.61	566.21	577.15	560.12
2	634.61	616.09	617.67	621.27	653.56	615.98	614.57	628.80	603.80	602.46	619.78	610.06	596.26
3	628.61	627.88	625.40	621.87	640.78	607.04	607.30	613.26	593.93	592.23	615.94	604.49	599.13
4	577.44	570.06	570.90	570.90	587.90	582.28	583.47	583.54	569.27	573.67	569.31	566.50	564.50
5	579.28	583.03	572.08	573.40	588.82	567.35	563.74	575.15	557.96	557.52	571.39	561.11	558.50
6	697.28	680.97	665.69	666.25	691.68	657.76	648.31	675.42	652.00	639.93	672.51	642.00	637.43
7	693.83	691.92	688.68	679.23	690.88	656.11	649.47	678.68	650.86	641.64	685.77	659.50	639.33
8	609.25	601.30	594.88	595.08	605.38	586.82	582.75	602.16	571.92	579.22	588.05	579.00	568.48
9	623.31	616.25	613.74	615.48	606.99	572.54	578.02	604.85	587.28	568.92	590.27	568.24	565.87
10	614.48	591.47	591.44	591.50	590.31	569.07	566.56	579.36	585.08	563.01	576.58	564.27	561.72
11	618.92	605.72	613.28	613.08	637.67	612.49	609.66	618.62	605.39	606.13	614.61	600.93	593.39
12	603.58	606.80	605.06	605.43	614.75	591.68	592.29	604.93	589.49	592.06	600.52	584.42	585.38
13	586.90	580.68	575.73	576.04	581.72	560.27	562.33	567.21	567.98	551.58	563.11	558.99	544.34
14	585.04	593.36	593.67	593.70	601.86	576.35	567.65	600.05	589.07	574.65	579.54	575.42	568.19
15	638.04	623.69	610.05	613.60	631.31	593.39	589.85	615.67	589.84	583.84	601.90	593.42	579.44
16	618.79	604.66	599.51	599.55	611.47	577.17	573.01	597.40	569.39	573.85	584.52	585.18	566.26
17	647.77	656.96	635.11	633.61	634.48	595.98	600.27	642.76	609.45	587.79	631.79	598.48	583.30
18	615.90	614.32	609.39	609.41	625.11	600.21	605.12	611.79	594.70	588.63	597.85	595.39	586.77
19	612.12	599.98	589.77	598.78	612.15	592.30	586.30	601.57	581.71	578.39	591.84	578.07	576.18
20	604.79	590.34	588.66	588.39	597.40	570.79	570.56	592.59	565.94	567.62	581.26	564.69	568.19
21	634.52	631.40	619.50	620.03	635.69	612.48	612.20	617.27	610.30	603.73	619.53	599.06	595.62
22	610.54	595.64	583.61	583.75	584.42	556.90	559.04	584.34	565.30	555.76	567.16	566.66	550.56
23	625.08	598.05	589.56	589.60	630.95	592.22	593.92	609.96	583.80	581.03	587.33	592.01	575.70
24	643.69	631.86	631.29	631.33	647.74	623.90	607.12	633.85	621.20	605.12	623.65	603.70	600.11
25	633.46	617.57	621.17	620.01	634.86	614.93	611.50	626.48	602.24	614.19	610.05	604.73	596.48
26	596.49	614.89	587.97	587.81	608.41	589.50	595.28	597.40	605.53	578.14	595.92	590.67	591.67
27	614.20	611.58	596.55	596.55	622.89	594.91	583.22	599.56	584.50	588.56	598.28	582.72	596.45
28	623.97	625.27	612.21	612.62	642.64	609.22	602.51	618.71	603.30	596.88	618.69	597.85	595.58
29	607.10	598.49	602.90	602.96	629.50	595.90	593.41	616.39	588.93	580.93	591.90	596.07	582.51
30	616.11	595.31	590.20	592.98	615.53	584.62	582.00	610.75	573.59	574.55	595.69	582.44	568.64
Avg.	620.70	612.00	605.51	606.12	623.00	594.87	591.94	610.27	591.43	585.42	600.36	589.44	581.87

Note: 1: do-nothing; 2: DET-NDP-D; 3: SP-NDP-D; 4: RO-NDP-D; 5: DET-SSD; 6: SP-SSD; 7: RO-SSD; 8: DET-NDP; 9: SP-NDP; 10: RO-NDP; 11: DET-NDP-SSD; 12: SP-NDP-SSD; 13: RO-NDP-SSD

Table I.2. Risk of 30 OD Demand Scenarios for Network 3.2 ($TSTT^{TARGET} = 575.206$)

Scenario	Problem (see note)												
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	2587.25	88.05	0.00	16.53	3346.83	115.17	0.00	593.25	0.00	0.00	0.00	3.77	0.00
2	3528.24	1671.23	1803.47	2122.05	6139.87	1662.79	1549.13	2871.96	817.81	742.78	1986.69	1214.45	443.13
3	2851.56	2774.49	2519.55	2177.63	4299.88	1013.47	1029.99	1448.19	350.76	289.85	1659.08	857.62	572.49
4	4.98	0.00	0.00	0.00	161.18	50.02	68.21	69.46	0.00	0.00	0.00	0.00	0.00
5	16.59	61.28	0.00	0.00	185.23	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
6	14902.31	11187.00	8187.25	8289.21	13567.20	6815.35	5344.11	10042.10	5896.72	4189.34	9468.28	4461.59	3872.17
7	14071.42	13622.30	12876.90	10821.10	13379.40	6545.28	5514.73	10706.60	5723.61	4413.11	12225.30	7104.73	4111.32
8	1158.99	680.81	386.87	395.04	910.37	134.81	56.94	726.34	0.00	16.11	164.97	14.37	0.00
9	2313.99	1684.31	1485.10	1622.11	1010.20	0.00	7.93	878.45	145.78	0.00	226.97	0.00	0.00
10	1542.29	264.44	263.46	265.47	228.06	0.00	0.00	17.28	97.41	0.00	1.90	0.00	0.00
11	1910.91	930.92	1449.65	1434.59	3902.20	1390.24	1186.98	1884.94	910.99	956.21	1552.96	661.91	330.79
12	805.03	998.20	891.48	913.61	1564.01	271.46	291.70	883.63	203.95	284.07	640.64	84.81	103.58
13	136.63	30.00	0.28	0.69	42.36	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
14	96.77	329.45	340.80	341.97	710.17	1.32	0.00	616.98	192.21	0.00	18.81	0.05	0.00
15	3947.86	2350.38	1213.87	1473.97	3147.10	330.54	214.40	1637.20	214.25	74.55	712.48	331.69	17.88
16	1899.91	867.39	590.64	592.59	1315.26	3.86	0.00	492.76	0.00	0.00	86.67	99.50	0.00
17	5265.10	6684.08	3588.16	3410.90	3513.08	431.51	628.07	4563.84	1172.46	158.41	3201.62	541.86	65.44
18	1656.33	1530.12	1168.51	1170.22	2490.85	625.17	894.81	1338.23	380.06	180.32	512.65	407.57	133.64
19	1362.64	613.82	212.01	555.49	1365.08	292.08	123.15	695.13	42.30	10.16	276.51	8.19	0.94
20	875.04	229.14	181.10	173.91	492.72	0.00	0.00	302.32	0.00	0.00	36.67	0.00	0.00
21	3517.68	3157.86	1961.88	2009.56	3657.81	1389.04	1368.62	1769.31	1231.65	813.67	1964.40	569.05	416.63
22	1248.21	417.38	70.56	72.96	84.94	0.00	0.00	83.48	0.00	0.00	0.00	0.00	0.00
23	2487.32	521.86	206.09	207.29	3107.42	289.60	350.29	1208.09	73.86	33.90	146.88	282.27	0.24
24	4690.33	3209.30	3144.85	3150.09	5261.42	2371.11	1018.61	3439.31	2115.60	894.65	2346.66	811.91	620.38
25	3393.65	1794.26	2112.82	2006.93	3557.97	1577.84	1317.23	2628.82	730.73	1519.73	1214.08	871.75	452.57
26	453.18	1574.89	163.03	158.76	1102.23	204.25	402.80	492.69	919.70	8.58	428.90	239.26	270.93
27	1520.22	1323.13	455.39	455.39	2273.68	388.41	64.16	593.16	86.33	178.40	532.60	56.40	451.49
28	2378.12	2505.90	1369.17	1399.93	4546.89	1156.73	745.33	1892.60	789.27	469.69	1890.71	512.90	414.90
29	1017.23	542.07	766.77	770.10	2948.20	428.31	331.33	1696.40	188.26	32.76	278.69	435.24	53.30
30	1673.06	404.14	224.90	315.95	1625.96	88.61	46.15	1263.20	0.00	0.00	419.77	52.29	0.00
Avg.	2777.09	2068.27	1587.82	1544.13	2997.92	919.23	751.82	1827.86	742.79	508.88	1399.83	654.11	411.06

Note: 1: do-nothing; 2: DET-NDP-D; 3: SP-NDP-D; 4: RO-NDP-D; 5: DET-SSD; 6: SP-SSD; 7: RO-SSD; 8: DET-NDP; 9: SP-NDP; 10: RO-NDP; 11: DET-NDP-SSD; 12: SP-NDP-SSD; 13: RO-NDP-SSD

Table I.3. Robust_Obj(1) of 30 OD Demand Scenarios for Network 3.2

Scenario	Problem (see note)												
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	3213.32	672.64	569.72	595.80	3979.89	701.11	566.86	1192.82	569.11	560.61	566.21	580.92	560.12
2	4162.85	2287.32	2421.15	2743.32	6793.44	2278.77	2163.70	3500.76	1421.61	1345.24	2606.47	1824.51	1039.39
3	3480.17	3402.37	3144.95	2799.50	4940.66	1620.52	1637.29	2061.45	944.69	882.09	2275.02	1462.11	1171.62
4	582.41	570.06	570.90	570.90	749.08	632.30	651.68	653.00	569.27	573.67	569.31	566.50	564.50
5	595.87	644.31	572.08	573.40	774.04	567.35	563.74	575.15	557.96	557.52	571.39	561.11	558.50
6	15599.59	11867.90	8852.94	8955.46	14258.90	7473.11	5992.42	10717.50	6548.72	4829.27	10140.80	5103.59	4509.60
7	14765.25	14314.20	13565.50	11500.30	14070.30	7201.39	6164.20	11385.30	6374.47	5054.74	12911.00	7764.22	4750.64
8	1768.24	1282.11	981.75	990.12	1515.75	721.62	639.69	1328.50	571.92	595.33	753.02	593.37	568.48
9	2937.30	2300.56	2098.85	2237.59	1617.19	572.54	585.95	1483.30	733.05	568.92	817.24	568.24	565.87
10	2156.77	855.90	854.90	856.97	818.36	569.07	566.56	596.64	682.49	563.01	578.48	564.27	561.72
11	2529.83	1536.63	2062.93	2047.67	4539.88	2002.73	1796.64	2503.57	1516.38	1562.34	2167.57	1262.84	924.18
12	1408.61	1605.00	1496.54	1519.04	2178.77	863.14	883.99	1488.56	793.43	876.13	1241.16	669.23	688.97
13	723.53	610.68	576.01	576.73	624.08	560.27	562.33	567.21	567.98	551.58	563.11	558.99	544.34
14	681.81	922.80	934.46	935.67	1312.02	577.67	567.65	1217.02	781.28	574.65	598.36	575.47	568.19
15	4585.90	2974.06	1823.92	2087.57	3778.40	923.92	804.25	2252.87	804.10	658.39	1314.38	925.11	597.32
16	2518.71	1472.05	1190.15	1192.14	1926.73	581.03	573.01	1090.16	569.39	573.85	671.19	684.68	566.26
17	5912.87	7341.04	4223.26	4044.51	4147.56	1027.49	1228.33	5206.61	1781.91	746.21	3833.41	1140.34	648.74
18	2272.23	2144.45	1777.90	1779.63	3115.97	1225.38	1499.93	1950.01	974.76	768.96	1110.50	1002.97	720.40
19	1974.76	1213.80	801.77	1154.26	1977.24	884.38	709.45	1296.71	624.01	588.55	868.35	586.26	577.12
20	1479.82	819.48	769.76	762.30	1090.12	570.79	570.56	894.91	565.94	567.62	617.93	564.69	568.19
21	4152.19	3789.26	2581.38	2629.60	4293.50	2001.52	1980.82	2386.58	1841.95	1417.40	2583.92	1168.11	1012.25
22	1858.74	1013.01	654.16	656.71	669.37	556.90	559.04	667.82	565.30	555.76	567.16	566.66	550.56
23	3112.40	1119.91	795.66	796.90	3738.37	881.82	944.21	1818.06	657.66	614.93	734.20	874.28	575.95
24	5334.02	3841.15	3776.14	3781.42	5909.16	2995.01	1625.73	4073.17	2736.80	1499.76	2970.31	1415.61	1220.49
25	4027.11	2411.83	2733.99	2626.93	4192.82	2192.77	1928.73	3255.30	1332.97	2133.92	1824.13	1476.48	1049.05
26	1049.67	2189.78	751.00	746.56	1710.63	793.74	998.07	1090.09	1525.23	586.72	1024.81	829.94	862.59
27	2134.42	1934.71	1051.93	1051.93	2896.57	983.33	647.37	1192.72	670.83	766.96	1130.89	639.11	1047.94
28	3002.09	3131.17	1981.38	2012.55	5189.53	1765.94	1347.83	2511.31	1392.57	1066.57	2509.40	1110.76	1010.47
29	1624.33	1140.56	1369.67	1373.06	3577.70	1024.21	924.73	2312.79	777.18	613.69	870.59	1031.31	635.81
30	2289.16	999.45	815.10	908.93	2241.49	673.23	628.15	1873.95	573.59	574.55	1015.46	634.73	568.64
Avg.	3397.80	2680.27	2193.33	2150.25	3620.92	1514.10	1343.76	2438.13	1334.22	1094.30	2000.19	1243.55	992.93

Note: 1: do-nothing; 2: DET-NDP-D; 3: SP-NDP-D; 4: RO-NDP-D; 5: DET-SSD; 6: SP-SSD; 7: RO-SSD; 8: DET-NDP; 9: SP-NDP; 10: RO-NDP; 11: DET-NDP-SSD; 12: SP-NDP-SSD; 13: RO-NDP-SSD

Table I.4. TSTT of 30 OD Demand Scenarios for Network 4.2

Scenario	Problem (see note)												
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	4201.11	4279.76	4262.46	4232.80	4175.65	3884.10	3819.77	3855.39	3894.83	3841.75	3809.93	4032.17	3767.09
2	4221.10	4210.54	4201.90	4209.72	4157.53	4012.97	4055.03	4016.71	3993.93	4020.86	4051.48	4042.17	3955.03
3	4482.88	4480.71	4467.58	4429.65	4109.11	4189.32	4072.43	4101.86	4196.63	4142.91	4184.30	4166.13	3954.20
4	4096.72	4088.45	4094.55	4096.08	3859.14	3811.34	3669.81	3959.62	3852.52	3656.75	3765.12	3839.95	3673.68
5	4204.69	4215.75	4222.87	4202.46	3996.66	3986.32	3944.42	4023.59	3967.48	3957.03	3953.05	4063.94	4182.33
6	3769.22	3769.80	3761.97	3765.97	3928.11	3521.83	3422.80	3783.79	3587.22	3528.58	3562.78	3708.00	3473.21
7	4168.25	4145.32	4160.76	4152.50	4021.59	3895.41	3843.60	3857.28	3862.25	3871.73	3722.45	3942.58	3757.60
8	4240.89	4275.36	4263.07	4249.24	3864.54	4082.61	3898.35	3905.90	4012.51	3879.95	4063.88	3923.62	3971.98
9	3984.40	3953.40	3961.47	3965.59	3814.71	3748.15	3653.93	3553.50	3735.03	3700.61	3747.63	3796.44	3709.20
10	4079.32	4056.90	4068.81	4081.87	3807.82	3792.23	3729.10	3770.17	3783.00	3777.58	3951.21	3901.48	3685.42
11	4083.11	4114.65	4077.72	4066.18	3812.22	3861.58	3968.25	3805.63	3900.34	3926.95	3880.45	3802.19	3842.30
12	4397.16	4387.98	4353.85	4390.83	3985.17	4013.93	3982.06	3995.19	4083.67	4000.12	3866.11	3928.30	3951.36
13	4212.58	4488.29	4307.20	4303.81	4125.46	3873.91	3970.45	4162.67	3899.09	3906.39	4398.30	3901.16	3891.48
14	4120.42	3702.59	3753.94	3810.92	3542.00	3833.04	3701.89	3452.66	3818.34	3765.62	3945.92	3735.29	3828.57
15	4225.55	4219.06	4230.83	4231.08	4069.94	4011.18	3982.42	4081.04	3992.99	3991.72	3987.02	3879.71	3989.63
16	3718.67	3740.25	3759.05	3747.14	3591.28	3511.37	3570.06	3610.69	3512.86	3568.84	3546.99	3811.63	3554.71
17	3734.66	3712.04	3740.21	3759.94	3427.11	3480.04	3469.59	3408.16	3433.21	3438.82	3449.48	3338.33	3469.38
18	4052.42	4022.58	4042.44	4056.27	3596.75	3703.25	3674.39	3699.40	3697.16	3615.68	3824.59	3665.79	3675.02
19	3947.47	3900.19	3918.48	3893.13	3502.53	3737.05	3581.68	3643.51	3740.01	3559.56	3748.37	3486.28	3691.50
20	4131.39	4138.28	4132.92	4109.48	3650.26	3796.25	3775.43	3863.22	3750.92	3761.87	3774.23	3867.93	3830.90
21	4258.07	4241.74	4259.88	4247.60	3985.49	3970.06	3916.51	4070.24	3984.79	3891.53	3851.66	3666.68	3861.87
22	3999.21	3918.54	3967.84	3962.04	3653.46	3732.31	3739.16	3640.61	3769.75	3727.95	3721.41	3552.62	3663.55
23	4137.78	4099.83	4126.41	4131.96	4032.34	3827.25	3807.51	3917.23	3825.85	3786.30	3842.65	3858.56	3811.05
24	3927.44	3948.17	3913.19	3956.36	3636.93	3662.22	3564.62	3623.29	3658.80	3593.19	3717.56	3899.40	3551.30
25	4108.82	4098.49	4044.80	4049.34	3834.16	3720.08	3753.50	3823.80	3697.87	3732.49	3910.93	3589.82	3652.73
26	3789.67	3785.92	3760.07	3779.98	3425.51	3467.97	3524.51	3683.80	3456.16	3424.74	3397.15	3526.21	3471.45
27	4108.24	4072.26	4081.50	4062.15	4547.93	4239.32	4151.06	3965.30	4235.23	4136.21	3976.10	3937.41	3881.41
28	3428.43	3399.43	3412.61	3420.45	3366.07	3244.88	3077.99	3503.69	3162.46	3273.00	3408.58	3098.90	3346.55
29	3849.36	3819.94	3829.74	3828.15	3526.13	3636.84	3624.60	4044.35	3586.24	3632.72	3551.51	3503.85	3527.47
30	3969.82	3967.19	3926.23	3932.34	3631.35	3725.40	3677.46	3680.96	3754.33	3694.12	3621.54	3648.00	3489.59
Avg.	4054.96	4041.78	4036.81	4037.50	3822.57	3799.07	3754.08	3816.78	3794.85	3760.19	3807.75	3770.48	3737.05

Note: 1: do-nothing; 2: DET-NDP-D; 3: SP-NDP-D; 4: RO-NDP-D; 5: DET-SSD; 6: SP-SSD; 7: RO-SSD; 8: DET-NDP; 9: SP-NDP; 10: RO-NDP; 11: DET-NDP-SSD; 12: SP-NDP-SSD; 13: RO-NDP-SSD

Table I.5. Risk of 30 OD Demand Scenarios for Network 4.2 ($TSTT^{TARGET} = 3628.43$)

Scenario	Problem (see note)												
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	327967.00	424234.00	401988.00	365258.00	299449.00	65364.90	36609.90	51510.30	70970.10	45506.40	32941.20	163002.00	19225.40
2	351258.00	338849.00	328872.00	337895.00	279944.00	147869.00	181991.00	150758.00	133589.00	153998.00	178973.00	171178.00	106670.00
3	730079.00	726384.00	704167.00	641959.00	231058.00	314596.00	197137.00	224138.00	322855.00	264695.00	308993.00	289118.00	106123.00
4	219293.00	211618.00	217270.00	218696.00	53228.30	33457.60	1712.10	109687.00	50216.00	802.16	18683.90	44739.70	2047.79
5	332070.00	344943.00	353362.00	329511.00	135594.00	128084.00	99852.10	156153.00	114955.00	107979.00	105380.00	189668.00	306810.00
6	19822.10	19986.20	17832.30	18917.90	89809.60	0.00	0.00	24136.00	0.00	0.00	0.00	6331.78	0.00
7	291407.00	267178.00	283371.00	274654.00	154572.00	71277.30	46296.60	52373.00	54670.90	59194.80	8839.71	98693.20	16686.10
8	375112.00	418517.00	402762.00	385405.00	55746.50	206277.00	72857.00	76987.90	147518.00	63264.10	189615.00	87137.30	118026.00
9	126718.00	105605.00	110915.00	113674.00	34699.40	14332.30	650.31	0.00	11362.70	5209.63	14208.10	28226.10	6524.51
10	203301.00	183589.00	193932.00	205610.00	32179.90	26830.70	10135.00	20089.50	23890.60	22246.00	104184.00	74558.60	3248.15
11	206734.00	236414.00	201864.00	191626.00	33779.60	54359.20	115478.00	31398.90	73932.60	89111.50	63513.50	30193.30	45739.90
12	590939.00	576909.00	526235.00	581257.00	127260.00	148608.00	125053.00	134511.00	207241.00	138152.00	56491.90	89923.20	104286.00
13	341228.00	739354.00	460732.00	456139.00	247043.00	60260.70	116979.00	285415.00	73255.30	77260.20	592706.00	74382.90	69196.50
14	242050.00	5499.99	15752.80	33302.10	0.00	41863.30	5396.90	0.00	36064.60	18819.80	100801.00	11419.50	40057.50
15	356549.00	348839.00	362883.00	363186.00	194927.00	146497.00	125307.00	204855.00	132902.00	131983.00	128589.00	63139.50	130464.00
16	8143.86	12504.80	17062.90	14091.70	0.00	0.00	0.00	0.00	0.00	0.00	0.00	33562.90	0.00
17	11284.80	6990.31	12494.70	17293.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
18	179766.00	155354.00	171404.00	183050.00	0.00	5597.78	2112.32	5036.27	4723.81	0.00	38479.40	1395.65	2170.16
19	101789.00	73851.10	84126.40	70067.30	0.00	11798.80	0.00	227.52	12449.80	0.00	14384.50	0.00	3977.26
20	252965.00	259946.00	254506.00	231405.00	476.51	28162.70	21610.20	55127.50	15003.60	17805.10	21256.40	57359.90	40994.40
21	396448.00	376151.00	398733.00	383377.00	127489.00	116711.00	82990.70	195196.00	126993.00	69223.90	49832.90	1463.02	54492.40
22	137478.00	84162.00	115200.00	111292.00	626.52	10790.80	12261.20	148.36	19972.40	9903.29	8645.02	0.00	1233.43
23	259440.00	222220.00	247987.00	253543.00	163147.00	39529.70	32070.50	83405.90	38974.30	24922.70	45889.10	52961.70	33350.40
24	89409.00	102235.00	81088.30	107541.00	72.34	1141.43	0.00	0.00	922.44	0.00	7944.75	73423.80	0.00
25	230772.00	220954.00	173367.00	177165.00	42324.90	8398.86	15642.20	38169.50	4821.72	10828.20	79807.30	0.00	590.50
26	25997.30	24802.10	17330.40	22967.90	0.00	0.00	0.00	3065.47	0.00	0.00	0.00	0.00	0.00
27	230218.00	196987.00	205274.00	188113.00	845474.00	373193.00	273139.00	113480.00	368208.00	257844.00	120873.00	95467.60	64000.60
28	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
29	48811.70	36674.90	40523.80	39888.20	0.00	70.68	0.00	172988.00	0.00	18.36	0.00	0.00	0.00
30	116547.00	114756.00	88684.80	92358.20	8.53	9402.86	2404.10	2759.05	15852.10	4315.40	0.00	383.12	0.00
Avg.	226786.56	227850.25	216324.01	213641.44	104963.64	68815.82	52589.50	73053.87	68711.47	52436.08	76367.72	57924.29	42530.47

Note: 1: do-nothing; 2: DET-NDP-D; 3: SP-NDP-D; 4: RO-NDP-D; 5: DET-SSD; 6: SP-SSD; 7: RO-SSD; 8: DET-NDP; 9: SP-NDP; 10: RO-NDP; 11: DET-NDP-SSD; 12: SP-NDP-SSD; 13: RO-NDP-SSD

Table I.6. *Robust_Obj(1)* of 30 OD Demand Scenarios for Network 4.2

Scenario	Problem (see note)												
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	332168	428513	406251	369491	303624	69249	40429.7	55365.7	74865	49348.1	36751.2	167035	22992.4
2	355479	343059	333074	342105	284101	151882	186046	154775	137583	158019	183025	175221	110625
3	734562	730865	708635	646388	235167	318785	201210	228239	327052	268838	313178	293284	110077
4	223390	215706	221365	222792	57087.4	37268.9	5381.91	113647	54068.5	4458.92	22449	48579.6	5721.47
5	336275	349159	357584	333714	139590	132071	103797	160177	118923	111936	109333	193732	310992
6	23591.3	23756	21594.2	22683.9	93737.7	3521.83	3422.8	27919.7	3587.22	3528.58	3562.78	10039.8	3473.21
7	295575	271324	287532	278807	158594	75172.7	50140.2	56230.2	58533.1	63066.5	12562.2	102636	20443.7
8	379353	422792	407025	389655	59611	210359	76755.3	80893.8	151530	67144.1	193679	91060.9	121998
9	130702	109559	114877	117640	38514.1	18080.5	4304.24	3553.5	15097.8	8910.24	17955.7	32022.5	10233.7
10	207380	187646	198001	209692	35987.7	30622.9	13864.1	23859.7	27673.6	26023.6	108135	78460.1	6933.57
11	210818	240528	205941	195692	37591.8	58220.8	119446	35204.5	77833	93038.5	67394	33995.5	49582.2
12	595336	581297	530588	585648	131245	152622	129035	138506	211324	142153	60358	93851.5	108237
13	345441	743843	465039	460443	251168	64134.6	120950	289577	77154.4	81166.6	597104	78284	73088
14	246170	9202.59	19506.8	37113	3542	45696.4	9098.8	3452.66	39883	22585.4	104747	15154.8	43886
15	360775	353058	367114	367417	198997	150508	129290	208936	136895	135974	132576	67019.3	134453
16	11862.5	16245.1	20821.9	17838.8	3591.28	3511.37	3570.06	3610.69	3512.86	3568.84	3546.99	37374.5	3554.71
17	15019.4	10702.3	16234.9	21053.9	3427.11	3480.04	3469.59	3408.16	3433.21	3438.82	3449.48	3338.33	3469.38
18	183819	159377	175447	187106	3596.75	9301.03	5786.71	8735.66	8420.97	3615.68	42304	5061.43	5845.18
19	105736	77751.3	88044.9	73960.4	3502.53	15535.8	3581.68	3871.04	16189.9	3559.56	18132.9	3486.28	7668.76
20	257096	264084	258639	235515	4126.77	31959	25385.7	58990.7	18754.5	21567	25030.7	61227.8	44825.3
21	400706	380392	402992	387625	131475	120681	86907.2	199266	130978	73115.5	53684.5	5129.7	58354.3
22	141477	88080.6	119168	115255	4279.98	14523.1	16000.4	3788.97	23742.1	13631.2	12366.4	3552.62	4896.98
23	263578	226320	252113	257675	167179	43357	35878.1	87323.2	42800.2	28709	49731.8	56820.3	37161.4
24	93336.4	106183	85001.4	111498	3709.27	4803.64	3564.62	3623.29	4581.24	3593.19	11662.3	77323.2	3551.3
25	234880	225052	177412	181215	46159.1	12118.9	19395.7	41993.3	8519.59	14560.7	83718.3	3589.82	4243.23
26	29786.9	28588	21090.5	26747.9	3425.51	3467.97	3524.51	6749.26	3456.16	3424.74	3397.15	3526.21	3471.45
27	234326	201059	209355	192175	850022	377432	277290	117446	372443	261980	124849	99405	67882
28	3428.43	3399.43	3412.61	3420.45	3366.07	3244.88	3077.99	3503.69	3162.46	3273	3408.58	3098.9	3346.55
29	52661	40494.8	44353.6	43716.3	3526.13	3707.51	3624.6	177033	3586.24	3651.08	3551.51	3503.85	3527.47
30	120517	118723	92611.1	96290.6	3639.88	13128.3	6081.57	6440.01	19606.4	8009.52	3621.54	4031.12	3489.59
Avg.	230841.50	231891.97	220360.80	217679.11	108786.14	72614.87	56343.65	76870.66	72506.32	56196.28	80175.50	61694.84	46267.46

Note: 1: do-nothing; 2: DET-NDP-D; 3: SP-NDP-D; 4: RO-NDP-D; 5: DET-SSD; 6: SP-SSD; 7: RO-SSD; 8: DET-NDP; 9: SP-NDP; 10: RO-NDP; 11: DET-NDP-SSD; 12: SP-NDP-SSD; 13: RO-NDP-SSD

Table I.7. TSTT Differences of 30 OD Demand Scenarios for Network 3.2

Scenario	D(13-1)	D(13-2)	D(13-3)	D(13-4)	D(13-5)	D(13-6)	D(13-7)	D(13-8)	D(13-9)	D(13-10)	D(13-11)	D(13-12)
1	-65.95	-24.47	-9.60	-19.15	-72.94	-25.82	-6.74	-39.44	-8.99	-0.49	-6.09	-17.03
2	-38.35	-19.83	-21.42	-25.02	-57.31	-19.73	-18.31	-32.54	-7.55	-6.20	-23.52	-13.80
3	-29.47	-28.75	-26.27	-22.74	-41.65	-7.91	-8.17	-14.13	5.20	6.90	-16.80	-5.36
4	-12.93	-5.55	-6.39	-6.39	-23.40	-17.78	-18.96	-19.04	-4.77	-9.17	-4.81	-1.99
5	-20.78	-24.54	-13.58	-14.90	-30.32	-8.85	-5.24	-16.65	0.54	0.98	-12.89	-2.61
6	-59.85	-43.54	-28.26	-28.82	-54.25	-20.33	-10.88	-37.98	-14.56	-2.50	-35.08	-4.57
7	-54.50	-52.60	-49.36	-39.91	-51.55	-16.78	-10.14	-39.35	-11.54	-2.31	-46.45	-20.17
8	-40.77	-32.82	-26.39	-26.60	-36.90	-18.34	-14.27	-33.68	-3.44	-10.74	-19.57	-10.52
9	-57.44	-50.38	-47.87	-49.61	-41.12	-6.66	-12.15	-38.97	-21.41	-3.05	-24.40	-2.37
10	-52.76	-29.75	-29.72	-29.78	-28.59	-7.35	-4.85	-17.65	-23.36	-1.29	-14.87	-2.55
11	-25.53	-12.32	-19.89	-19.69	-44.28	-19.10	-16.27	-25.23	-12.00	-12.74	-21.22	-7.54
12	-18.19	-21.42	-19.68	-20.05	-29.37	-6.30	-6.90	-19.55	-4.10	-6.68	-15.13	0.97
13	-42.55	-36.34	-31.39	-31.70	-37.37	-15.93	-17.98	-22.87	-23.64	-7.24	-18.77	-14.65
14	-16.86	-25.17	-25.48	-25.51	-33.67	-8.17	0.53	-31.86	-20.88	-6.47	-11.36	-7.24
15	-58.60	-44.25	-30.61	-34.16	-51.87	-13.95	-10.41	-36.23	-10.41	-4.41	-22.46	-13.98
16	-52.53	-38.40	-33.25	-33.29	-45.21	-10.91	-6.75	-31.14	-3.13	-7.59	-18.26	-18.92
17	-64.47	-73.67	-51.81	-50.31	-51.18	-12.68	-16.97	-59.47	-26.15	-4.50	-48.49	-15.19
18	-29.14	-27.56	-22.62	-22.65	-38.35	-13.44	-18.35	-25.02	-7.94	-1.87	-11.08	-8.63
19	-35.94	-23.80	-13.59	-22.60	-35.98	-16.12	-10.13	-25.39	-5.53	-2.22	-15.66	-1.89
20	-36.60	-22.15	-20.47	-20.20	-29.21	-2.60	-2.37	-24.40	2.26	0.58	-13.07	3.50
21	-38.90	-35.78	-23.88	-24.42	-40.07	-16.86	-16.58	-21.65	-14.68	-8.11	-23.91	-3.44
22	-59.98	-45.08	-33.05	-33.19	-33.86	-6.34	-8.48	-33.78	-14.75	-5.20	-16.60	-16.10
23	-49.38	-22.35	-13.86	-13.90	-55.25	-16.52	-18.22	-34.26	-8.10	-5.33	-11.63	-16.31
24	-43.58	-31.74	-31.17	-31.22	-47.63	-23.79	-7.01	-33.74	-21.09	-5.00	-23.54	-3.59
25	-36.98	-21.09	-24.69	-23.53	-38.38	-18.45	-15.02	-30.00	-5.76	-17.71	-13.57	-8.25
26	-4.83	-23.22	3.69	3.86	-16.74	2.17	-3.61	-5.74	-13.87	13.53	-4.25	0.99
27	-17.74	-15.13	-0.09	-0.09	-26.44	1.54	13.24	-3.11	11.96	7.89	-1.83	13.74
28	-28.40	-29.69	-16.63	-17.05	-47.06	-13.64	-6.93	-23.14	-7.72	-1.30	-23.11	-2.28
29	-24.59	-15.98	-20.39	-20.45	-47.00	-13.40	-10.90	-33.89	-6.42	1.58	-9.39	-13.56
30	-47.47	-26.67	-21.56	-24.34	-46.89	-15.98	-13.36	-42.11	-4.95	-5.91	-27.05	-13.80
Avg.	-38.84	-30.13	-23.64	-24.25	-41.13	-13.00	-10.07	-28.40	-9.56	-3.55	-18.50	-7.57
SD	16.68	13.76	12.69	11.79	11.75	6.82	6.94	11.41	8.91	6.11	10.78	7.74
Half-width of 95%CI	6.23	5.14	4.74	4.40	4.39	2.55	2.59	4.26	3.33	2.28	4.03	2.89
Lower	-45.06	-35.27	-28.38	-28.65	-45.51	-15.55	-12.66	-32.66	-12.88	-5.83	-22.52	-10.46
Upper	-32.61	-25.00	-18.91	-19.84	-36.74	-10.45	-7.48	-24.14	-6.23	-1.27	-14.47	-4.68

Note: 1: do-nothing; 2: DET-NDP-D; 3: SP-NDP-D; 4: RO-NDP-D; 5: DET-SSD; 6: SP-SSD; 7: RO-SSD; 8: DET-NDP; 9: SP-NDP; 10: RO-NDP; 11: DET-NDP-SSD; 12: SP-NDP-SSD; 13: RO-NDP-SSD

Table I.8. Risk Differences of 30 OD Demand Scenarios for Network 3.2

Scenario	D(13-1)	D(13-2)	D(13-3)	D(13-4)	D(13-5)	D(13-6)	D(13-7)	D(13-8)	D(13-9)	D(13-10)	D(13-11)	D(13-12)
1	-2587.25	-88.05	0.00	-16.53	-3346.83	-115.17	0.00	-593.25	0.00	0.00	0.00	-3.77
2	-3085.11	-1228.10	-1360.34	-1678.92	-5696.74	-1219.66	-1106.00	-2428.83	-374.68	-299.65	-1543.56	-771.32
3	-2279.07	-2202.00	-1947.06	-1605.14	-3727.39	-440.98	-457.50	-875.70	221.74	282.64	-1086.59	-285.13
4	-4.98	0.00	0.00	0.00	-161.18	-50.02	-68.21	-69.46	0.00	0.00	0.00	0.00
5	-16.59	-61.28	0.00	0.00	-185.23	0.00	0.00	0.00	0.00	0.00	0.00	0.00
6	-11030.14	-7314.83	-4315.08	-4417.04	-9695.03	-2943.18	-1471.94	-6169.93	-2024.55	-317.17	-5596.11	-589.42
7	-9960.10	-9510.98	-8765.58	-6709.78	-9268.08	-2433.96	-1403.41	-6595.28	-1612.29	-301.79	-8113.98	-2993.41
8	-1158.99	-680.81	-386.87	-395.04	-910.37	-134.81	-56.94	-726.34	0.00	-16.11	-164.97	-14.37
9	-2313.99	-1684.31	-1485.10	-1622.11	-1010.20	0.00	-7.93	-878.45	-145.78	0.00	-226.97	0.00
10	-1542.29	-264.44	-263.46	-265.47	-228.06	0.00	0.00	-17.28	-97.41	0.00	-1.90	0.00
11	-1580.12	-600.13	-1118.86	-1103.80	-3571.41	-1059.45	-856.19	-1554.15	-580.20	-625.42	-1222.17	-331.12
12	-701.44	-894.61	-787.89	-810.02	-1460.43	-167.87	-188.12	-780.05	-100.36	-180.48	-537.06	18.77
13	-136.63	-30.00	-0.28	-0.69	-42.36	0.00	0.00	0.00	0.00	0.00	0.00	0.00
14	-96.77	-329.45	-340.80	-341.97	-710.17	-1.32	0.00	-616.98	-192.21	0.00	-18.81	-0.05
15	-3929.98	-2332.50	-1195.99	-1456.09	-3129.22	-312.65	-196.51	-1619.32	-196.37	-56.66	-694.60	-313.80
16	-1899.91	-867.39	-590.64	-592.59	-1315.26	-3.86	0.00	-492.76	0.00	0.00	-86.67	-99.50
17	-5199.66	-6618.64	-3522.72	-3345.46	-3447.64	-366.07	-562.62	-4498.40	-1107.02	-92.97	-3136.18	-476.42
18	-1522.69	-1396.48	-1034.87	-1036.58	-2357.21	-491.54	-761.18	-1204.59	-246.42	-46.69	-379.01	-273.94
19	-1361.70	-612.88	-211.06	-554.54	-1364.14	-291.14	-122.20	-694.19	-41.35	-9.21	-275.57	-7.25
20	-875.04	-229.14	-181.10	-173.91	-492.72	0.00	0.00	-302.32	0.00	0.00	-36.67	0.00
21	-3101.05	-2741.23	-1545.25	-1592.93	-3241.18	-972.41	-951.99	-1352.68	-815.02	-397.04	-1547.77	-152.42
22	-1248.21	-417.38	-70.56	-72.96	-84.94	0.00	0.00	-83.48	0.00	0.00	0.00	0.00
23	-2487.07	-521.62	-205.85	-207.05	-3107.18	-289.35	-350.04	-1207.85	-73.62	-33.66	-146.63	-282.03
24	-4069.96	-2588.93	-2524.48	-2529.72	-4641.05	-1750.74	-398.24	-2818.94	-1495.23	-274.27	-1726.29	-191.53
25	-2941.07	-1341.69	-1660.25	-1554.36	-3105.40	-1125.27	-864.66	-2176.25	-278.16	-1067.16	-761.51	-419.18
26	-182.25	-1303.96	107.90	112.17	-831.30	66.68	-131.87	-221.76	-648.77	262.34	-157.97	31.66
27	-1068.74	-871.65	-3.90	-3.90	-1822.20	63.07	387.33	-141.67	365.15	273.09	-81.12	395.09
28	-1963.23	-2091.00	-954.27	-985.03	-4131.99	-741.83	-330.43	-1477.70	-374.38	-54.79	-1475.81	-98.01
29	-963.93	-488.77	-713.47	-716.80	-2894.90	-375.01	-278.03	-1643.10	-134.96	20.54	-225.39	-381.94
30	-1673.06	-404.14	-224.90	-315.95	-1625.96	-88.61	-46.15	-1263.20	0.00	0.00	-419.77	-52.29
Avg.	-2366.03	-1657.21	-1176.76	-1133.07	-2586.86	-508.17	-340.76	-1416.80	-331.73	-97.82	-988.77	-243.05
SD	2551.44	2264.11	1784.52	1489.43	2410.94	750.45	458.71	1668.41	557.79	264.71	1783.14	568.21
Half-width of 95%CI	952.62	845.34	666.28	556.10	900.16	280.19	171.27	622.93	208.26	98.83	665.76	212.15
Lower	-3318.65	-2502.55	-1843.03	-1689.17	-3487.02	-788.36	-512.03	-2039.72	-539.99	-196.65	-1654.53	-455.20
Upper	-1413.42	-811.88	-510.48	-576.97	-1686.70	-227.98	-169.49	-793.87	-123.47	1.02	-323.01	-30.89

Note: 1: do-nothing; 2: DET-NDP-D; 3: SP-NDP-D; 4: RO-NDP-D; 5: DET-SSD; 6: SP-SSD; 7: RO-SSD; 8: DET-NDP; 9: SP-NDP; 10: RO-NDP; 11: DET-NDP-SSD; 12: SP-NDP-SSD; 13: RO-NDP-SSD

Table I.9. Robust_Obj(1) Differences of 30 OD Demand Scenarios for Network 3.2

Scenario	D(13-1)	D(13-2)	D(13-3)	D(13-4)	D(13-5)	D(13-6)	D(13-7)	D(13-8)	D(13-9)	D(13-10)	D(13-11)	D(13-12)
1	-2653.20	-112.52	-9.60	-35.68	-3419.77	-140.99	-6.74	-632.70	-8.99	-0.49	-6.09	-20.80
2	-3123.46	-1247.93	-1381.76	-1703.93	-5754.05	-1239.38	-1124.31	-2461.37	-382.22	-305.85	-1567.08	-785.12
3	-2308.55	-2230.75	-1973.33	-1627.88	-3769.04	-448.90	-465.67	-889.83	226.93	289.53	-1103.40	-290.49
4	-17.91	-5.55	-6.39	-6.39	-184.58	-67.79	-87.17	-88.49	-4.77	-9.17	-4.81	-1.99
5	-37.37	-85.81	-13.58	-14.90	-215.54	-8.85	-5.24	-16.65	0.54	0.98	-12.89	-2.61
6	-11089.99	-7358.30	-4343.34	-4445.86	-9749.30	-2963.51	-1482.82	-6207.90	-2039.12	-319.67	-5631.20	-593.99
7	-10014.61	-9563.56	-8814.86	-6749.66	-9319.66	-2450.75	-1413.56	-6634.66	-1623.83	-304.10	-8160.36	-3013.58
8	-1199.76	-713.63	-413.26	-421.64	-947.27	-153.14	-71.21	-760.02	-3.44	-26.85	-184.54	-24.88
9	-2371.43	-1734.69	-1532.98	-1671.72	-1051.32	-6.66	-20.08	-917.43	-167.18	-3.05	-251.37	-2.37
10	-1595.05	-294.19	-293.18	-295.25	-256.65	-7.35	-4.85	-34.92	-120.77	-1.29	-16.77	-2.55
11	-1605.65	-612.45	-1138.75	-1123.49	-3615.70	-1078.55	-872.46	-1579.39	-592.20	-638.16	-1243.39	-338.66
12	-719.64	-916.03	-807.57	-830.07	-1489.80	-174.17	-195.02	-799.59	-104.47	-187.16	-552.19	19.74
13	-179.19	-66.34	-31.67	-32.39	-79.74	-15.93	-17.98	-22.87	-23.64	-7.24	-18.77	-14.65
14	-113.62	-354.62	-366.28	-367.48	-743.83	-9.48	0.53	-648.83	-213.09	-6.47	-30.17	-7.28
15	-3988.58	-2376.74	-1226.60	-1490.25	-3181.08	-326.60	-206.93	-1655.55	-206.78	-61.07	-717.06	-327.79
16	-1952.45	-905.79	-623.89	-625.88	-1360.47	-14.77	-6.75	-523.90	-3.13	-7.59	-104.93	-118.42
17	-5264.13	-6692.30	-3574.52	-3395.77	-3498.82	-378.75	-579.59	-4557.87	-1133.17	-97.47	-3184.67	-491.60
18	-1551.83	-1424.05	-1057.50	-1059.23	-2395.57	-504.98	-779.53	-1229.61	-254.36	-48.56	-390.10	-282.57
19	-1397.64	-636.68	-224.65	-577.14	-1400.12	-307.26	-132.33	-719.59	-46.89	-11.43	-291.23	-9.14
20	-911.63	-251.29	-201.57	-194.11	-521.93	-2.60	-2.37	-326.72	2.26	0.58	-49.74	3.50
21	-3139.94	-2777.01	-1569.13	-1617.35	-3281.25	-989.27	-968.57	-1374.33	-829.70	-405.15	-1571.67	-155.86
22	-1308.19	-462.45	-103.60	-106.15	-118.81	-6.34	-8.48	-117.26	-14.75	-5.20	-16.60	-16.10
23	-2536.45	-543.97	-219.71	-220.95	-3162.43	-305.88	-368.26	-1242.12	-81.72	-38.99	-158.26	-298.33
24	-4113.53	-2620.66	-2555.65	-2560.93	-4688.67	-1774.52	-405.24	-2852.68	-1516.31	-279.27	-1749.82	-195.12
25	-2978.06	-1362.78	-1684.94	-1577.88	-3143.77	-1143.72	-879.68	-2206.25	-283.92	-1084.87	-775.08	-427.43
26	-187.08	-1327.19	111.59	116.03	-848.04	68.85	-135.48	-227.50	-662.64	275.87	-162.22	32.65
27	-1086.48	-886.77	-3.99	-3.99	-1848.63	64.61	400.57	-144.78	377.11	280.98	-82.95	408.83
28	-1991.62	-2120.70	-970.91	-1002.08	-4179.06	-755.47	-337.36	-1500.84	-382.10	-56.10	-1498.93	-100.29
29	-988.52	-504.75	-733.86	-737.25	-2941.89	-388.40	-288.93	-1676.98	-141.38	22.11	-234.78	-395.50
30	-1720.53	-430.81	-246.46	-340.29	-1672.85	-104.59	-59.51	-1305.31	-4.95	-5.91	-446.82	-66.09
Avg.	-2404.87	-1687.34	-1200.40	-1157.32	-2627.99	-521.17	-350.83	-1445.20	-341.29	-101.37	-1007.26	-250.62
SD	2560.92	2273.39	1792.31	1495.76	2418.41	754.07	462.14	1674.88	562.15	269.23	1791.62	571.87
Half-width of 95%CI	956.16	848.80	669.18	558.46	902.95	281.54	172.54	625.34	209.89	100.52	668.93	213.51
Lower	-3361.03	-2536.14	-1869.58	-1715.78	-3530.94	-802.71	-523.38	-2070.54	-551.18	-201.89	-1676.19	-464.13
Upper	-1448.71	-838.54	-531.22	-598.86	-1725.04	-239.63	-178.29	-819.86	-131.40	-0.85	-338.34	-37.10

Note: 1: do-nothing; 2: DET-NDP-D; 3: SP-NDP-D; 4: RO-NDP-D; 5: DET-SSD; 6: SP-SSD; 7: RO-SSD; 8: DET-NDP; 9: SP-NDP; 10: RO-NDP; 11: DET-NDP-SSD; 12: SP-NDP-SSD; 13: RO-NDP-SSD

Table I.10. TSTT Differences of 30 OD Demand Scenarios for Network 4.2

Scenario	D(13-1)	D(13-2)	D(13-3)	D(13-4)	D(13-5)	D(13-6)	D(13-7)	D(13-8)	D(13-9)	D(13-10)	D(13-11)	D(13-12)
1	-434.02	-512.67	-495.37	-465.71	-408.56	-117.01	-52.68	-88.30	-127.74	-74.66	-42.84	-265.08
2	-266.07	-255.51	-246.87	-254.69	-202.50	-57.94	-100.00	-61.68	-38.90	-65.83	-96.45	-87.14
3	-528.68	-526.51	-513.38	-475.45	-154.91	-235.12	-118.23	-147.66	-242.43	-188.71	-230.10	-211.93
4	-423.04	-414.77	-420.87	-422.40	-185.46	-137.66	3.87	-285.94	-178.84	16.93	-91.44	-166.27
5	-22.36	-33.42	-40.54	-20.13	185.67	196.01	237.91	158.74	214.85	225.30	229.28	118.39
6	-296.01	-296.59	-288.76	-292.76	-454.90	-48.62	50.41	-310.58	-114.01	-55.37	-89.57	-234.79
7	-410.65	-387.72	-403.16	-394.90	-263.99	-137.81	-86.00	-99.68	-104.65	-114.13	35.15	-184.98
8	-268.91	-303.38	-291.09	-277.26	107.44	-110.63	73.63	66.08	-40.53	92.03	-91.90	48.36
9	-275.20	-244.20	-252.27	-256.39	-105.51	-38.95	55.27	155.70	-25.83	8.59	-38.43	-87.24
10	-393.90	-371.48	-383.39	-396.45	-122.40	-106.81	-43.68	-84.75	-97.58	-92.16	-265.79	-216.06
11	-240.81	-272.35	-235.42	-223.88	30.08	-19.28	-125.95	36.67	-58.04	-84.65	-38.15	40.11
12	-445.80	-436.62	-402.49	-439.47	-33.81	-62.57	-30.70	-43.83	-132.31	-48.76	85.25	23.06
13	-321.10	-596.81	-415.72	-412.33	-233.98	17.57	-78.97	-271.19	-7.61	-14.91	-506.82	-9.68
14	-291.85	125.98	74.63	17.65	286.57	-4.47	126.68	375.91	10.23	62.95	-117.35	93.28
15	-235.92	-229.43	-241.20	-241.45	-80.31	-21.55	7.21	-91.41	-3.36	-2.09	2.61	109.92
16	-163.96	-185.54	-204.34	-192.43	-36.57	43.34	-15.35	-55.98	41.85	-14.13	7.72	-256.92
17	-265.28	-242.66	-270.83	-290.56	42.27	-10.66	-0.21	61.22	36.17	30.56	19.90	131.05
18	-377.40	-347.56	-367.42	-381.25	78.27	-28.23	0.63	-24.38	-22.14	59.34	-149.57	9.23
19	-255.97	-208.69	-226.98	-201.63	188.97	-45.55	109.82	47.99	-48.51	131.94	-56.87	205.22
20	-300.49	-307.38	-302.02	-278.58	180.64	34.65	55.47	-32.32	79.98	69.03	56.67	-37.03
21	-396.20	-379.87	-398.01	-385.73	-123.62	-108.19	-54.64	-208.37	-122.92	-29.66	10.21	195.19
22	-335.66	-254.99	-304.29	-298.49	10.09	-68.76	-75.61	22.94	-106.20	-64.40	-57.86	110.93
23	-326.73	-288.78	-315.36	-320.91	-221.29	-16.20	3.54	-106.18	-14.80	24.75	-31.60	-47.51
24	-376.14	-396.87	-361.89	-405.06	-85.63	-110.92	-13.32	-71.99	-107.50	-41.89	-166.26	-348.10
25	-456.09	-445.76	-392.07	-396.61	-181.43	-67.35	-100.77	-171.07	-45.14	-79.76	-258.20	62.91
26	-318.22	-314.47	-288.62	-308.53	45.94	3.48	-53.06	-212.35	15.29	46.71	74.30	-54.76
27	-226.83	-190.85	-200.09	-180.74	-666.52	-357.91	-269.65	-83.89	-353.82	-254.80	-94.69	-56.00
28	-81.88	-52.88	-66.06	-73.90	-19.52	101.67	268.56	-157.14	184.09	73.55	-62.03	247.65
29	-321.89	-292.47	-302.27	-300.68	1.34	-109.37	-97.13	-516.88	-58.77	-105.25	-24.04	23.62
30	-480.23	-477.60	-436.64	-442.75	-141.76	-235.81	-187.87	-191.37	-264.74	-204.53	-131.95	-158.41
Avg.	-317.91	-304.73	-299.76	-300.45	-85.51	-62.02	-17.03	-79.72	-57.80	-23.13	-70.69	-33.43
SD	111.62	151.20	129.45	125.80	202.58	103.64	111.56	166.03	116.96	100.58	132.44	155.09
Half-width of 95%CI	41.67	56.45	48.33	46.97	75.64	38.70	41.65	61.99	43.67	37.55	49.45	57.91
Lower	-359.58	-361.18	-348.09	-347.42	-161.15	-100.72	-58.68	-141.71	-101.47	-60.69	-120.14	-91.34
Upper	-276.23	-248.28	-251.43	-253.48	-9.88	-23.33	24.63	-17.73	-14.13	14.42	-21.25	24.47

Note: 1: do-nothing; 2: DET-NDP-D; 3: SP-NDP-D; 4: RO-NDP-D; 5: DET-SSD; 6: SP-SSD; 7: RO-SSD; 8: DET-NDP; 9: SP-NDP; 10: RO-NDP; 11: DET-NDP-SSD; 12: SP-NDP-SSD; 13: RO-NDP-SSD

Table I.11. Risk Differences of 30 OD Demand Scenarios for Network 4.2

Scenario	D(13-1)	D(13-2)	D(13-3)	D(13-4)	D(13-5)	D(13-6)	D(13-7)	D(13-8)	D(13-9)	D(13-10)	D(13-11)	D(13-12)
1	-308741.6	-405008.6	-382762.6	-346032.6	-280223.6	-46139.5	-17384.5	-32284.9	-51744.7	-26281.0	-13715.8	-143776.6
2	-244588.0	-232179.0	-222202.0	-231225.0	-173274.0	-41199.0	-75321.0	-44088.0	-26919.0	-47328.0	-72303.0	-64508.0
3	-623956.0	-620261.0	-598044.0	-535836.0	-124935.0	-208473.0	-91014.0	-118015.0	-216732.0	-158572.0	-202870.0	-182995.0
4	-217245.2	-209570.2	-215222.2	-216648.2	-51180.5	-31409.8	335.7	-107639.2	-48168.2	1245.6	-16636.1	-42691.9
5	-25260.0	-38133.0	-46552.0	-22701.0	171216.0	178726.0	206957.9	150657.0	191855.0	198831.0	201430.0	117142.0
6	-19822.1	-19986.2	-17832.3	-18917.9	-89809.6	0.0	0.0	-24136.0	0.0	0.0	0.0	-6331.8
7	-274720.9	-250491.9	-266684.9	-257967.9	-137885.9	-54591.2	-29610.5	-35686.9	-37984.8	-42508.7	7846.4	-82007.1
8	-257086.0	-300491.0	-284736.0	-267379.0	62279.5	-88251.0	45169.0	41038.1	-29492.0	54761.9	-71589.0	30888.7
9	-120193.5	-99080.5	-104390.5	-107149.5	-28174.9	-7807.8	5874.2	6524.5	-4838.2	1314.9	-7683.6	-21701.6
10	-200052.9	-180340.9	-190683.9	-202361.9	-28931.8	-23582.6	-6886.9	-16841.4	-20642.5	-18997.9	-100935.9	-71310.5
11	-160994.1	-190674.1	-156124.1	-145886.1	11960.3	-8619.3	-69738.1	14341.0	-28192.7	-43371.6	-17773.6	15546.6
12	-486653.0	-472623.0	-421949.0	-476971.0	-22974.0	-44322.0	-20767.0	-30225.0	-102955.0	-33866.0	47794.1	14362.8
13	-272031.5	-670157.5	-391535.5	-386942.5	-177846.5	8935.8	-47782.5	-216218.5	-4058.8	-8063.7	-523509.5	-5186.4
14	-201992.5	34557.5	24304.7	6755.4	40057.5	-1805.8	34660.6	40057.5	3992.9	21237.7	-60743.5	28638.0
15	-226085.0	-218375.0	-232419.0	-232722.0	-64463.0	-16033.0	5157.0	-74391.0	-2438.0	-1519.0	1875.0	67324.5
16	-8143.9	-12504.8	-17062.9	-14091.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	-33562.9
17	-11284.8	-6990.3	-12494.7	-17293.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
18	-177595.8	-153183.8	-169233.8	-180879.8	2170.2	-3427.6	57.8	-2866.1	-2553.7	2170.2	-36309.2	774.5
19	-97811.7	-69873.8	-80149.1	-66090.0	3977.3	-7821.5	3977.3	3749.7	-8472.5	3977.3	-10407.2	3977.3
20	-211970.6	-218951.6	-213511.6	-190410.6	40517.9	12831.7	19384.2	-14133.1	25990.8	23189.3	19738.0	-16365.5
21	-341955.6	-321658.6	-344240.6	-328884.6	-72996.6	-62218.6	-28498.3	-140703.6	-72500.6	-14731.5	4659.5	53029.4
22	-136244.6	-82928.6	-113966.6	-110058.6	606.9	-9557.4	-11027.8	1085.1	-18739.0	-8669.9	-7411.6	1233.4
23	-226089.6	-188869.6	-214636.6	-220192.6	-129796.6	-6179.3	1279.9	-50055.5	-5623.9	8427.7	-12538.7	-19611.3
24	-89409.0	-102235.0	-81088.3	-107541.0	-72.3	-1141.4	0.0	0.0	-922.4	0.0	-7944.8	-73423.8
25	-230181.5	-220363.5	-172776.5	-176574.5	-41734.4	-7808.4	-15051.7	-37579.0	-4231.2	-10237.7	-79216.8	590.5
26	-25997.3	-24802.1	-17330.4	-22967.9	0.0	0.0	0.0	-3065.5	0.0	0.0	0.0	0.0
27	-166217.4	-132986.4	-141273.4	-124112.4	-781473.4	-309192.4	-209138.4	-49479.4	-304207.4	-193843.4	-56872.4	-31467.0
28	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
29	-48811.7	-36674.9	-40523.8	-39888.2	0.0	-70.7	0.0	-172988.0	0.0	-18.4	0.0	0.0
30	-116547.0	-114756.0	-88684.8	-92358.2	-8.5	-9402.9	-2404.1	-2759.1	-15852.1	-4315.4	0.0	-383.1
Avg.	-184256.1	-185319.8	-173793.5	-171111.0	-62433.2	-26285.4	-10059.0	-30523.4	-26181.0	-9905.6	-33837.3	-15393.8
SD	141052.1	174748.0	147490.0	141950.3	160396.5	77290.9	61911.5	69095.2	79294.4	62146.2	111492.4	57789.7
HW of 95%CI	52663.8	65244.6	55067.5	52999.2	59886.3	28857.6	23115.6	25797.7	29605.7	23203.2	41627.3	21576.6
Lower	-236919.9	-250564.4	-228861.0	-224110.1	-122319.5	-55143.0	-33174.6	-56321.1	-55786.7	-33108.8	-75464.5	-36970.4
Upper	-131592.3	-120075.1	-118726.1	-118111.8	-2546.9	2572.3	13056.5	-4725.7	3424.7	13297.6	7790.0	6182.8

Note: 1: do-nothing; 2: DET-NDP-D; 3: SP-NDP-D; 4: RO-NDP-D; 5: DET-SSD; 6: SP-SSD; 7: RO-SSD; 8: DET-NDP; 9: SP-NDP; 10: RO-NDP; 11: DET-NDP-SSD; 12: SP-NDP-SSD; 13: RO-NDP-SSD

Table I.12. Robust_Obj(1) Differences of 30 OD Demand Scenarios for Network 4.2

Scenario	D(13-1)	D(13-2)	D(13-3)	D(13-4)	D(13-5)	D(13-6)	D(13-7)	D(13-8)	D(13-9)	D(13-10)	D(13-11)	D(13-12)
1	-309175.6	-405520.6	-383258.6	-346498.6	-280631.6	-46256.6	-17437.3	-32373.3	-51872.6	-26355.7	-13758.8	-144042.6
2	-244854.0	-232434.0	-222449.0	-231480.0	-173476.0	-41257.0	-75421.0	-44150.0	-26958.0	-47394.0	-72400.0	-64596.0
3	-624485.0	-620788.0	-598558.0	-536311.0	-125090.0	-208708.0	-91133.0	-118162.0	-216975.0	-158761.0	-203101.0	-183207.0
4	-217668.5	-209984.5	-215643.5	-217070.5	-51365.9	-31547.4	339.6	-107925.5	-48347.0	1262.6	-16727.5	-42858.1
5	-25283.0	-38167.0	-46592.0	-22722.0	171402.0	178921.0	207195.0	150815.0	192069.0	199056.0	201659.0	117260.0
6	-20118.1	-20282.8	-18121.0	-19210.7	-90264.5	-48.6	50.4	-24446.5	-114.0	-55.4	-89.6	-6566.6
7	-275131.3	-250880.3	-267088.3	-258363.3	-138150.3	-54729.0	-29696.5	-35786.5	-38089.4	-42622.8	7881.5	-82192.3
8	-257355.0	-300794.0	-285027.0	-267657.0	62387.0	-88361.0	45242.7	41104.2	-29532.0	54853.9	-71681.0	30937.1
9	-120468.3	-99325.3	-104643.3	-107406.3	-28280.4	-7846.8	5929.5	6680.2	-4864.1	1323.5	-7722.0	-21788.8
10	-200446.4	-180712.4	-191067.4	-202758.4	-29054.1	-23689.3	-6930.5	-16926.1	-20740.0	-19090.0	-101201.4	-71526.5
11	-161235.8	-190945.8	-156358.8	-146109.8	11990.4	-8638.6	-69863.8	14377.7	-28250.8	-43456.3	-17811.8	15586.7
12	-487099.0	-473060.0	-422351.0	-477411.0	-23008.0	-44385.0	-20798.0	-30269.0	-103087.0	-33916.0	47879.0	14385.5
13	-272353.0	-670755.0	-391951.0	-387355.0	-178080.0	8953.4	-47862.0	-216489.0	-4066.4	-8078.6	-524016.0	-5196.0
14	-202284.0	34683.4	24379.2	6773.0	40344.0	-1810.4	34787.2	40433.3	4003.0	21300.6	-60861.0	28731.2
15	-226322.0	-218605.0	-232661.0	-232964.0	-64544.0	-16055.0	5163.0	-74483.0	-2442.0	-1521.0	1877.0	67433.7
16	-8307.8	-12690.4	-17267.2	-14284.1	-36.6	43.3	-15.3	-56.0	41.8	-14.1	7.7	-33819.8
17	-11550.0	-7232.9	-12765.5	-17584.5	42.3	-10.7	-0.2	61.2	36.2	30.6	19.9	131.1
18	-177973.8	-153531.8	-169601.8	-181260.8	2248.4	-3455.9	58.5	-2890.5	-2575.8	2229.5	-36458.8	783.8
19	-98067.2	-70082.5	-80376.1	-66291.6	4166.2	-7867.0	4087.1	3797.7	-8521.1	4109.2	-10464.1	4182.5
20	-212270.7	-219258.7	-213813.7	-190689.7	40698.5	12866.3	19439.6	-14165.4	26070.8	23258.3	19794.6	-16402.5
21	-342351.7	-322037.7	-344637.7	-329270.7	-73120.7	-62326.7	-28552.9	-140911.7	-72623.7	-14761.2	4669.8	53224.6
22	-136580.0	-83183.6	-114271.0	-110358.0	617.0	-9626.1	-11103.4	1108.0	-18845.1	-8734.2	-7469.4	1344.4
23	-226416.6	-189158.6	-214951.6	-220513.6	-130017.6	-6195.6	1283.3	-50161.8	-5638.8	8452.4	-12570.4	-19658.9
24	-89785.1	-102631.7	-81450.1	-107946.7	-158.0	-1252.3	-13.3	-72.0	-1029.9	-41.9	-8111.0	-73771.9
25	-230636.8	-220808.8	-173168.8	-176971.8	-41915.9	-7875.7	-15152.5	-37750.1	-4276.4	-10317.5	-79475.1	653.4
26	-26315.5	-25116.6	-17619.1	-23276.5	45.9	3.5	-53.1	-3277.8	15.3	46.7	74.3	-54.8
27	-166444.0	-133177.0	-141473.0	-124293.0	-782140.0	-309550.0	-209408.0	-49564.0	-304561.0	-194098.0	-56967.0	-31523.0
28	-81.9	-52.9	-66.1	-73.9	-19.5	101.7	268.6	-157.1	184.1	73.6	-62.0	247.7
29	-49133.5	-36967.3	-40826.1	-40188.8	1.3	-180.0	-97.1	-173505.5	-58.8	-123.6	-24.0	23.6
30	-117027.4	-115233.4	-89121.5	-92801.0	-150.3	-9638.7	-2592.0	-2950.4	-16116.8	-4519.9	-132.0	-541.5
Avg.	-184574.0	-185624.5	-174093.3	-171411.6	-62518.7	-26347.4	-10076.2	-30603.2	-26238.9	-9928.8	-33908.0	-15427.4
SD	141123.2	174858.6	147577.9	142033.2	160563.0	77376.7	61994.9	69214.9	79386.9	62227.2	111608.9	57899.3
HW of 95%CI	52690.3	65285.9	55100.3	53030.1	59948.5	28889.7	23146.7	25842.3	29640.2	23233.4	41670.8	21617.5
Lower	-237264.4	-250910.4	-229193.6	-224441.8	-122467.2	-55237.1	-33222.9	-56445.5	-55879.1	-33162.2	-75578.8	-37044.9
Upper	-131883.7	-120338.6	-118993.0	-118381.5	-2570.2	2542.3	13070.5	-4760.8	3401.4	13304.6	7762.7	6190.1

Note: 1: do-nothing; 2: DET-NDP-D; 3: SP-NDP-D; 4: RO-NDP-D; 5: DET-SSD; 6: SP-SSD; 7: RO-SSD; 8: DET-NDP; 9: SP-NDP; 10: RO-NDP; 11: DET-NDP-SSD; 12: SP-NDP-SSD; 13: RO-NDP-SSD

References

1. Abdelfatah, A. S. and H. S. Mahmassani. "System optimal time-dependent path assignment and signal timing in traffic network," *Transportation Research Record* 1645, p 185-193, 1998.
2. Abdulaal, M., and L. J. LeBlanc. Continuous Equilibrium Network Design Models. *Transportation Research B*, Vol. 13, 1979, pp.19-32.
3. Aho, A. V., J. E. Hopcroft and J. D. Ullman. *Data Structures and Algorithms*, Addison-Wesley, Reading, MA, 1985.
4. Allsop, R. E. Some possibilities for using traffic control to influence trip distribution and route choice. D. J. Buckley (ed.), *Proceedings of the Sixth International Symposium on Transportation and Traffic Theory*, p 345-373, 1974.
5. Allsop, R. E. and J. A. Charlesworth. Traffic in a single-controlled road network: an example of different signal timings inducing different routings. *Traffic Engineering and Control*, Vol. 18, p 262-264, 1977.
6. Baker, J. E. Reducing bias and inefficiency in the selection algorithm. In *Genetic Algorithms and Their Applications: Proceedings of the Second International Conference on Genetic Algorithms*, Massachusetts Institute of Technology, Hillsdale, N.J., July 28-31, 1987.
7. Bard, J. An Efficient Point Algorithm for a Linear Two-Stage Optimization Problem. *Operations Research*, Vol. 31, 1983, pp. 670-684.
8. Bard, J. F. *Practical Bilevel Optimization Algorithms and Applications*. Kluwer Academic Publishers, 1998.
9. Battiti, R. and G. Tecchiolli. The Reactive Tabu Search. *ORSA Journal on Computing*, Vol.6, No.2, 1994, p.126-140.
10. Battiti, R. and G. Tecchiolli. The continuous reactive tabu search: blending combinatorial optimization and stochastic search for global optimization. *Annals of Operations Research*, No.63, 1996, p.153.
11. Bialas, W. F. and M. H. Karwan. On-Two-Level Optimization. *IEEE Transactions Automatic Control*, Vol. AC-27, No.1, pp.211-214, 1982.
12. Boender, C.G. and H.E. Romeijn. Stochastic methods. R.Horst and P.M. Pardalos (ed.), *Handbook of Global Optimization*, pp.829-869. Kluwer, the Netherlands, 1995.
13. Boyce, D. E. Urban Transportation Network Equilibrium and Design Models: Recent Achievements and Future Prospectives. *Environment and Planning A*, Vol. 16, 1984, pp. 1445-1474.
14. Bronson, R. *The Theory and Problems of Matrix Operations: Schaum's Outline Series*. McGraw-Hill, Inc., New York, 1989.
15. Brooks, S.H. A discussion of random methods for seeking maxima. *Operations Research*, Vol.6, pp.244-251, 1958.
16. Ceylan, H. and M.G.H. Bell. Traffic signal timing optimization based on genetic algorithm approach, including drivers' routing. *Transportation Research Part B*, Vol. 38, 2004, 329-342.

17. Chang, E. J. *Time-varying intermodal person trip assignment*, Ph.D. Dissertation, Northwestern University, 2004.
18. Charlesworth, J. A. The calculation of mutually consistent signal settings and traffic assignment for a signal-controlled road network. T. Sasaki and T. Yamaoka (ed.), *Proceedings of the Seventh International Symposium on Transportation and Traffic Theory*, p 545-569, 1977.
19. Chen, M., and A. S. Alfa. A Network Design Algorithm Using a Stochastic Incremental Traffic Assignment Approach. *Transportation Science*, Vol. 25, No. 3, 1991, pp. 215-224.
20. Chen, H.K. and C.F. Hsueh. "Combining Signal Timing Plan and Dynamic Traffic Assignment," *Presented at 76th Annual meeting of the Transportation Research Board*, Washington, D.C., 1997.
21. Chiou, S-W. Modeling a continuous network design problem for signal-controlled road network. *Presented at 84th Annual Meeting of the Transportation Research Board*, Washington, D.C., 2005.
22. Cipriani, E. and G. Fusco. Combined signal setting design and traffic assignment problem. *European Journal of Operational Research*, Vol. 155, p 569-583, 2004.
23. Daganzo, C. F. The Cell Transmission Model: A Simple Dynamic Representation of Highway Traffic Consistent with the Hydrodynamic Theory. *Transportation Research B*, Vol. 28, No. 4, 1994, pp. 269-287.
24. Daganzo, C. F. The Cell Transmission Model, Part II: Network Traffic. *Transportation Research B*, Vol. 29, No. 2, 1995, pp.79-93.
25. Dantzig, G. B. *Linear Programming and Extensions*, Princeton, NJ, 1963.
26. Dantzig, G. B., R. P. Harvey, Z. F. Landsowne, D. W. Robinson, and S. F. Maier. Formulating and Solving the Network Design Problem by Decomposition. *Transportation Research B*, Vol. 13, 1979, pp. 5-17.
27. Davis, G. A. Exact Local Solution of the Continuous Network Design Problem via Stochastic User Equilibrium Assignment. *Transportation Research B*, Vol. 28, 1994, pp. 61-75.
28. DeJong, K.A. An Analysis of the Behavior of a Class of Genetic Adaptive Systems. (Doctoral Dissertation, University of Michigan). *Dissertation Abstracts International* 36 (10), 5140B. (University Microfilms No. 76-9381), 1975.
29. Devore, J.L. *Probability and Statistics for Engineering and Sciences*, Duxbury Thomson Learning, 5th Edition, 2000.
30. Dorigo, M. and T. Stützle, *Ant Colony Optimization*, MIT Press, 2004.
31. Escudero, L.F., P.V. Kamesam, A. King and R.J.-B. Wets. "Production Planning Via Scenario Modeling," *Annals of Operations Research*, Vol.43, 311-335, 1993.
32. Fortuny-Amat J. and B. McCarl. A Representation and Economic Interpretation of a Two-Level Programming Problem. *Journal of the Operational Research Society*, Vol. 32, pp.783-792, 1981.
33. Friesz, T. L. Transportation Network Equilibrium, Design and Aggregation: Key Developments and Research Opportunities. *Transportation Research A*, Vol. 19, 1985, pp. 413-427.

34. Friesz, T. L., H. Cho, N. Mehta, R. Tobin, and G. Anandalingam. A Simulated Annealing Approach to the Network Design Problem with Variational Inequality Constraints. *Transportation Science*, Vol. 26, No. 1, 1992, pp. 18-26.
35. Gartner, N. H. Area traffic control and network equilibrium. M. A. Florian (ed.), *Traffic Equilibrium Methods: Proceedings of the International Symposium*, p 274-297, 1974.
36. Gartner, N. H. Area traffic control and network equilibrium. M. A. Florian (ed.), *Traffic Equilibrium Methods*, p 274-297, Springer-Verlag, New York, 1976.
37. Gartner, N.H. and C. Stamatiadis. "Integration of Dynamic Traffic Assignment with Real-Time Traffic Adaptive Control System," *Transportation Research Record* 1644, 150-156, 1997.
38. Gartner N. H., S. B. Gershwin, J. D. Little, and P. Ross. Pilot study of computer-based urban traffic management. *Transportation Research Part B*, Vol.14, No. 1/2, p 203-217, 1980.
39. Glover, F. "Tabu Search - Part I", *ORSA Journal on Computing*, Vol. 1, No. 3, pp 190-206, 1989.
40. Glover, F. "Tabu Search - Part II", *ORSA Journal on Computing*, Vol. 2, No. 1, pp 4-32, 1990.
41. Glover, F. and M. Laguna. *Tabu Search*. Kluwer Academic Publishers, Norwell, MA, USA, 1997.
42. Goldberg, D.E. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, MA, 1989.
43. Grefenstette, J.J. *A User's Guide to GENESIS Version 5.0*. October, 1990. <http://www.aic.nrl.navy.mil/galist/src/>
44. Gutierrez, G.J., P. Kouvelis and A.A. Kurawarwala. "Robustness Approach to Uncapacitated Network Design Problems," *European Journal of Operational Research*, Vol.94, No.2, 362-376, 1996.
45. Hearn, D.W., S. Lawphongpanich and S. Nguyen. "Convex programming formulations of the asymmetric traffic assignment problem," *Transportation Research Part B*, Vol.18, No.2, 357-365, 1984.
46. Hoang, H. H. Topological Optimization of Networks: A Non-Linear Mixed Integer Model Employing Generalized Benders Decomposition. *IEEE Transactions on Automated Control*, Vol. 27, 1982, pp. 164-169.
47. Holland, J. H. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, 1975.
48. Hooke, R., and T. A. Jeeves. Direct Search Solution of Numerical and Statistical Problems. *J. Ass. Comp. Mach*, Vol. 8, 1961, pp. 212-229.
49. Infanger, G. "Monte Carlo (Importance) Sampling With a Benders Decomposition Algorithm for Stochastic Linear Programs," *Annals of Operations Research*, Vol.39, 69-95, 1992.
50. Janson, B. N. Network Design Effects of Dynamic Traffic Assignment. *Journal of Transportation Engineering*, Vol. 121, No. 1, 1995, pp. 1-13.
51. Jeon, K., S. Ukkusuri and S. T. Waller. Heuristic Approach for Discrete Network Design Problem Accounting for Dynamic Traffic Assignment Conditions:

- Formulations, Solution Methodologies, Implementations and Computational Experiences. *Presented at 84th Annual Meeting of the Transportation Research Board*, Washington, D.C., 2005.
52. Kalantari, B. and J.B. Rosen. "Penalty for zero-one equivalent problem," *Mathematical Programming*, Vol.24, 229-232, 1982.
 53. Kall, P., and S. W. Wallace. *Stochastic Programming*. John Wiley & Sons, 1994.
 54. Karoonsoontawong, A. and S.T. Waller. A Comparison of System- and User-Optimal Stochastic Dynamic Network Design Models Using Monte Carlo Bounding Techniques. *Journal of the Transportation Research Board*, No. 1923, 2005, p 91-102.
 55. Karoonsoontawong, A. and S.T. Waller. Dynamic Continuous Network Design Problem: Linear Bi-level Programming and Metaheuristic Approaches. *Forthcoming in Transportation Research Record*, 2006.
 56. Keeney, R.L. and H. Raiffa. *Decisions With Multiple Objectives*, John Wiley, New York, 1976.
 57. Kirkpatrick, S., C.D. Gelatt and M.P. Vecchi. Optimization by Simulated Annealing. *Science* 220, 671-680, 1983.
 58. LeBlanc, L. J. An Algorithm for the Discrete Network Design Problem. *Transportation Science*, Vol. 9, 1975, pp. 183-199.
 59. LeBlanc, L. J., and D. Boyce. A Bi-Level Programming Algorithm for the Exact Solution of the Network Design Problem with User-Optimal Traffic Flows. *Transportation Research B*, Vol. 20, 1986, pp. 259-265.
 60. LeBlanc, L. J., and M. Abdulaal. An Efficient Dual Approach to the Urban Road Network Design Problem. *Computers and Mathematics with Applications*, Vol. 5, 1979, pp. 11-19.
 61. LeBlanc, L. J., and M. Abdulaal. A Comparison of the User-Optimum Versus System-Optimum Traffic Assignment in Transportation Network Design. *Transportation Research B*, Vol. 18, 1984, pp.115-121.
 62. Lee, C. *Combined traffic signal control and traffic assignment: algorithms, implementation and numerical results*. Ph.D. Dissertation, The University of Texas at Austin, Austin, Texas, 1998.
 63. Li, Y., S. T. Waller, and A. K. Ziliaskopoulos. A Decomposition Method for System-Optimal Dynamic Traffic Assignment. *Presented at 80th Annual Meeting of the Transportation Research Board*, Washington, D.C., 2000.
 64. Li, Y., S. T. Waller, and A. K. Ziliaskopoulos. A Decomposition Scheme for System-Optimal Dynamic Traffic Assignment Models. *Networks and Spatial Economics*, Vol. 3, 2003, 441-455.
 65. Lighthill, M.J. and J.B. Whitham. On kinematic waves, I: Flow modeling in long rivers; II: A theory of traffic flow on long crowded roads. *Proceedings of Royal Society A* 229, 1955, p.281-345.
 66. Magnanti, T. L., and R. T. Wong. Network Design and Transportation Planning: Models and Algorithms. *Transportation Science*, Vol. 18, No. 1, 1984, pp. 1-55.

67. Mak, W., D. P. Morton, R. K. Wood. Monte Carlo Bounding Techniques for Determining Solution Quality in Stochastic Programs. *Operations Research Letters*, Vol. 24, 1999, pp. 47-56.
68. Marcotte, P. Network Optimization with Continuous Control Parameters. *Transportation Science*, Vol. 17, 1983, pp. 181-197.
69. Marcotte, P. A Note on a Bilevel Programming Algorithm by Leblanc and Boyce. *Transportation Research B*, Vol. 22, 1988, pp. 233-237.
70. Markowitz, H. *Portfolio Selection, Efficiency Diversification of Investments*. Cowles Foundation Monograph 16, Yale University Press, New Haven, Conn., 1959 (second edition, Basil Blackwell, Cambridge, 1991).
71. Meng, Q., H. Yang, and M. G. H. Bell. An Equivalent Continuously Differentiable Model and a Locally Convergent Algorithm for the Continuous Network Design Problem. *Transportation Research B*, Vol. 35, 2001, pp. 83-105.
72. Metropolis, N., A. Rosenbluth, M. Rosenbluth, A. Teller and E. Teller. Equation of State Calculation by Fast Computing Machines. *Journal Chemical Physics* 21. 1087-1091, 1953.
73. Michalewicz, Z. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer Verlag, 3rd edition, 1996.
74. Morton, D. P. *Stochastic Optimization*, ORI 391Q Class Notes, The University of Texas at Austin, Austin, Texas, Spring 2004.
75. Mouskos, K. *A Taboo-Search Based Approach for Network Design*. Ph.D. dissertation. University of Texas at Austin, 1991.
76. Mulvey, J. M., R. J. Vanderbei, and S. A. Zenios. Robust Optimization of Large Scale Systems. *Operations Research*, Vol. 43, No. 2, p 264-281, 1995.
77. Munoz, L., S. Xiaotian, D. Sun, G. Gomes, R. Horowitz. Methodological Calibration of the Cell Transmission Model. *Proceeding of the 2004 American Control Conference*, Boston, Massachusetts, June 30 – July 2, 2004.
78. Murty, K. G. Solving the Fixed Charge Problem by Ranking the Extreme Points. *Operations Research*, XVI, pp.268-279, 1968.
79. Paraskevopoulos, D., E. Karakitsos and B. Rustem. "Robust Capacity Planning Under Uncertainty," *Management Science*, Vol.37, 787-800, 1991.
80. Park, B. *Development of genetic algorithm-based signal optimization program for oversaturated intersections*. Ph.D. Dissertation, Texas A&M University, 1998.
81. Patriksson, M., and R. T. Rockafellar. A Mathematical Model and Descent Algorithm for Bilevel Traffic Management. *Transportation Science*, Vol. 36, No. 3, 2002, pp. 271-291.
82. Peeta, S. and A.K. Ziliaskopoulos. "Foundations of dynamic traffic assignment: The past, the present and the future," *Networks and Spatial Economics*, Vol.1, No.3, 233-265, 2001.
83. Powell, M. J. An Efficient Method for Finding the Minimum of a Function of Several Variables without Using Derivatives. *Brit. Computer J*, Vol. 9, 1964, pp. 155-162.

84. Press, W.H., B.P. Flannery, S.A. Teukolsky and W.T. Vetterling. *Numerical Recipes: The Art of Scientific Computing (2nd edition)*. Cambridge University Press, New York, 1992.
85. Ran, B. and D.E. Boyce. "A link-based variational inequality formulation of ideal dynamic user-optimal route choice problem," *Transportation Research Part C*, Vol. 4, No. 1, 1-12, 1996.
86. Richards, P.I. Shockwaves on the highway. *Operations Research*, Vol.4, 1956, p.42-51.
87. Serra, R. and G. Zanarini. *Complex Systems and Cognitive Processes*, Springer-Verlag, Berlin, 1990.
88. Sheffi, Y. "Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Models," Englewood Cliffs, N.J., Prentice-Hall, 1985.
89. Shuster, H.G. *Deterministic Chaos*, Physik-Verlag, Weinheim, 1984.
90. Smith, M.J. "The existence and calculation of traffic equilibria," *Transportation Research Part B*, Vol.17, No.4, 291-303, 1983.
91. Smith, M.J. and M.O. Ghali. The dynamics of traffic assignment and traffic control: a theoretical study. *Transportation research 24B*, 1990, 409-422.
92. Smith, M.J. and T. Van Vuren. Traffic Equilibrium with responsive traffic control. *Transportation Science*, Vol. 27, No.2, 1993, 118-132.
93. Sun, W. and K. C. Mouskos. Network-wide traffic responsive signal control in urban environments. *Proceedings of the Joint Conference on Information Sciences*, v 5, n 1, *Proceedings of the Fifth Joint Conference on Information Sciences, JCIS 2000, Volume 1*, 2000, p 734-737.
94. Suwansirikul, C., T. L. Friesz, and R. L. Tobin. Equilibrium Decomposed Optimization: A Heuristic for the Continuous Equilibrium Network Design Problem. *Transportation Science*, Vol. 21, 1987, pp. 254-263.
95. Tan, H-N., S. B. Gershwin, and M. Athans. *Hybrid optimization in urban traffic networks*, Laboratory for Information and Decision Systems Technical Report DOT-TSC-RSPA-79-7, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1979.
96. Teklu, F. *Optimizing Traffic Control Signals Considering Routing: A Genetic Algorithm Approach*, Master's Thesis, Institute for Transport Studies, University of Leeds, UK, 2003.
97. Ukkusuri, S. V. S. K. *Linear Programs for the user Optimal Dynamic Traffic Assignment Problem*. Master's Thesis, University of Illinois at Urbana-Champaign, 2002.
98. Ukkusuri, S. V. S. K., A. Karoonsoontawong, and S. T. Waller. A Stochastic Dynamic User Optimal Network Design Model Accounting for Demand Uncertainty. *Proceedings of the International Conference of Transportations Systems Planning and Operations (TRANSPRO)*, Madras, India, Feb. 18-20, 2004.
99. Ukkusuri, S. V. S. K., and S. T. Waller. Linear Programming Models for the User and System Optimal Dynamic Network Design Problem: formulations, Implementations and Comparisons. *Presented at 83rd Annual Meeting of the Transportation Research Board*, Washington, D.C., 2004.

100. Vanderbilt, D. and S. G. Louie. A Monte Carlo Simulated Annealing Approach to Optimization over Continuous Variables. *Journal of Computational Physics* 56, 259-271, 1984.
101. Varia, H.R. and S.L. Dhingra. "Dynamic Optimal Traffic Assignment and Signal Time Optimization Using Genetic Algorithms," *Computer-Aided Civil and Infrastructure Engineering*, Vol. 19, 2004, p 260-273.
102. Von Stackelberg, H. *The Theory of the Market Economy*, Oxford University Press, Oxford, 1952.
103. Von Hohenbalken, B. "Simplicial decomposition in nonlinear programming algorithms," *Mathematical Programming*, Vol.13, 49-68, 1977.
104. Waller, S. T. *Optimization and Control of Stochastic Dynamic Transportation Systems: Formulations, Solution Methodologies, and Computational Experience*. Ph.D. Dissertation, Northwestern University, December 2000.
105. Waller, S.T., K.C. Mouskos, D. Kamaryiannis and A.K. Ziliaskopoulos, and A Linear Model for Continuous Network Design Problem. *Computer-Aided Civil and Infrastructure Engineering*, Vol. 21, No. 5, July, 2006, p 334-345.
106. Waller, S. T., and A. K. Ziliaskopoulos. Stochastic Dynamic Network Design Problem. *Journal of the Transportation Research Board*, No. 1771, TRB, National Research Council, Washington, D.C., 2001, pp.106-113.
107. Wardrop, John G. "Some Theoretical Aspects of Road Traffic Research," *Proceedings of the Institute of Civil Engineers, Part II* 1 (2), 325-378, 1952.
108. Wong, S. C. and C. Yang. An iterative group-based signal optimization scheme for traffic equilibrium networks. *Journal of Advanced Transportation*, Vol. 33, No. 2, 1999, p 201-217.
109. Yang, H., and M. G. H. Bell. Models and Algorithms for Road Network Design: A Review and Some New Developments. *Transport Reviews*, Vol. 18, No. 3, 1998, pp. 257-278.
110. Ziliaskopoulos, A. K. A Linear Programming Model for the Single Destination System Optimum Dynamic Traffic Assignment Problem. *Transportation Science*, Vol. 34, No. 1, 2000, pp. 37-49.
111. Ziliaskopoulos, A.K. and H.S. Mahmassani. (1994) Time-dependent, shortest-path algorithm for real-time intelligent vehicle highway system applications. *Transportation Research Record*, No. 1408, 94-100.
112. Ziliaskopoulos, A.K. and S. Lee. A cell transmission based assignment-simulation model for integrated freeway/surface street systems. *Transportation Research Record*, 1701, 1996, 12-23.
113. Ziliaskopoulos, A.K. and S.T. Waller. An Internet-based geographic information system that integrates data, models and users for transportation applications. *Transportation Research Part C* 8, 2000, 427-444.
114. Ziyoun and Yifan. A reserve capacity model of optimal signal control with user-equilibrium route choice. *Transportation Research Part B*, Vol. 36, p 313-323, 2002.

Vita

Ampol Karoonsoontawong was born in Bangkok, Thailand on March 11, 1976, the son of Somsak Karoonsoontawong and Anchalee Karoonsoontawong. After completing his work at Saint Gabriel's College, Bangkok, Thailand in 1993, he entered Chulalongkorn University in Bangkok, Thailand. He received the degree of Bachelor of Engineering from Chulalongkorn University in May 1997. After that, he was employed as a civil engineer at the Thai Engineering Consultant for a year. During the following two years, he worked as a governmental official at the Office of Accelerated Rural Development. In August 2000, he received an International Road Federation fellowship, and entered the Graduate School at The University of Texas at Austin in Transportation Engineering program. He finished his Master of Science degree in May of 2002, and subsequently pursued a Ph.D. in Transportation Engineering at the same institution.

Permanent Address: 3/84 Soi Ladphrao 31, Jatujak, Bangkok, 10900, Thailand

This dissertation was typed by the author.